Multi-Scale Rendering of Scratched Materials using a Structured SV-BRDF Model

Boris Raymond, Gael Guennebaud, Pascal Barla

Multi-Scale Rendering of Scratched Materials using a Structured SV-BRDF Model

Boris Raymond, Gaël Guennebaud, Pascal Barla
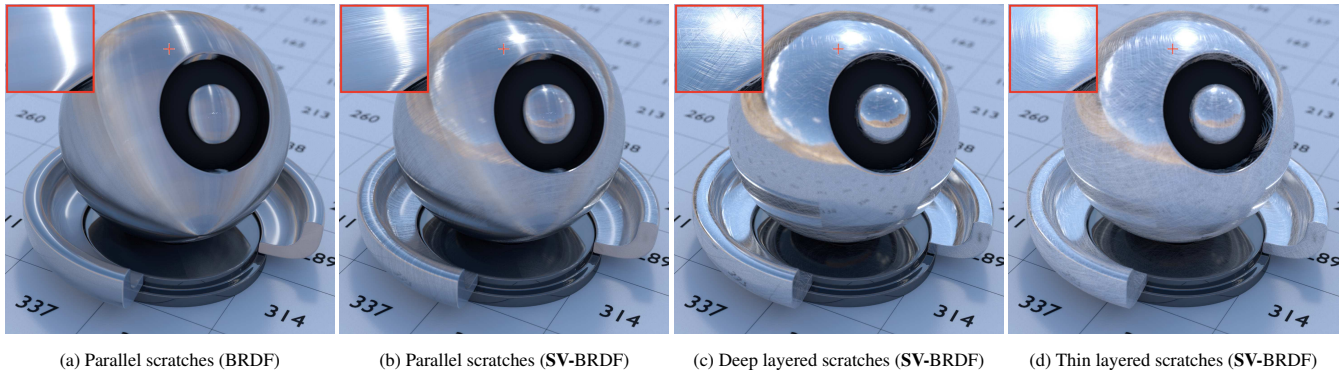Inria - Bordeaux University - CNRS

(a) Parallel scratches (BRDF)   (b) Parallel scratches (**SV**-BRDF)   (c) Deep layered scratches (**SV**-BRDF)   (d) Thin layered scratches (**SV**-BRDF)

Figure 1: *Our approach is tailored to the high angular and spatial frequencies exhibited by scratched materials. In (a) we show only angular variations via our scratch BRDF model, which imitates the appearance of brushed metal. A more realistic look is obtained in (b) by incorporating spatial variations thanks to our SV-BRDF model. Changing the profile and distribution of scratches in (c,d) yields different material appearances. Our model reproduces the characteristic look of scratched metals at multiple scales (insets show 3D camera zooms). This includes directionally-smeared reflections in (a,b), patterned highlights in (b), glint lines in (c,d) and hazy environment reflections in (d).*

Abstract

We introduce a Spatially-Varying BRDF model tailored to the multi-scale rendering of scratched materials such as metals, plastics or finished woods. Our approach takes advantage of the regular structure of scratch distributions to achieve high performance without compromising visual quality. We provide users with controls over the profile, micro-BRDF, density and orientation of scratches, while updating our material model at interactive rates. The BRDF for a single scratch is simulated using an optimized 2D ray-tracer and compactly stored in a three-component 2D texture. In contrast to existing models, our approach takes into account all inter-reflections inside a scratch, including Fresnel effects. At render time, the SV-BRDF for the scratch distribution under a pixel or ray footprint is obtained by linear combination of individual scratch BRDFs. We show how to evaluate it using both importance and light sampling, in direct and global illumination settings.

Concepts: •**Computing methodologies → Reflectance modeling;**

1 Introduction

Research on material appearance is pushing forward the development of ever-sophisticated and physically-plausible models. Traditionally in computer graphics, opaque materials are represented using Bidirectional Reflectance Distribution Functions (BRDF). For a homogeneous material, a single BRDF instance might be used to render an entire object. However, real-world materials are seldom,

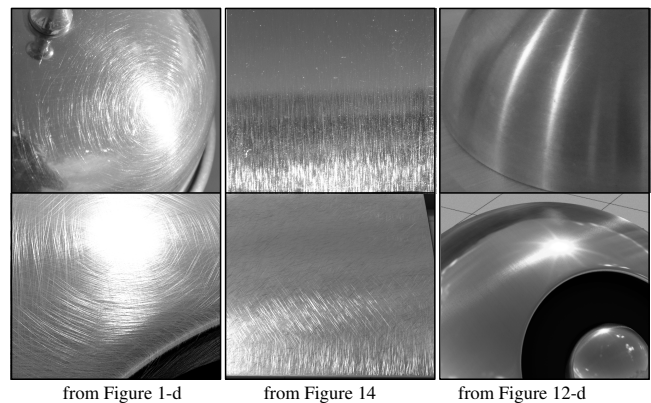from Figure 1-d   from Figure 14   from Figure 12-d

Figure 2: *Photographs of scratched materials (top row) used as inspiration for some of our results (bottom row). Scratches appear as glint lines (left) at moderate distances, while at farther distances, they produce patterned highlight silhouettes (middle) as well as directionally-smeared environment reflections (right).*

if ever, perfectly homogeneous: they often exhibit a lot of spatial micro-variations, in either reflectance or geometric details.

In this work, we focus on materials exhibiting many tiny scratches, as found mostly on metals, but also plastics or finished woods. The scratches may be due to polished or deteriorated objects, and are usually distributed all over the surface. As shown in Figure 2, they produce a number of interesting visual effects that are visible even from afar: glint lines around highlights, patterned highlight silhouettes, and directionally-smeared reflections.

As recently demonstrated by Yan et al. [2014], homogeneous BRDF models are not adapted in these cases; hence the modeling and rendering of high-frequency surface details remains a challenging problem. For extreme close-ups, scratches are individually visible in all their details. When getting farther away, reflections from individual scratches may still remain visible even though they are much smaller than a pixel: they lead to glint lines. At a distance, the distribution of scratches still has a visible impact on appear-

ance: it modulates highlight silhouettes and smears environment reflections. Scratched materials thus require a BRDF model that is not only spatially-varying, but also multi-scale.

The key insight of our work is to exploit the *regular micro-structure* of scratched materials. Individual scratches are mostly one-dimensional, and many of them are distributed parallel to each other. Complex scratch distributions (Figure 2-left) exhibit a combination of parallel scratches in different orientations, whereas simpler ones (Figure 2-right) only consist of a single set of orderly parallel scratches. In the following, we call a scratch *layer* a set of similar parallel scratches; a scratch *distribution* becomes a combination of one or more layers. This is not to be confused with the concept of layered materials (e.g., [Jakob et al. 2014a]): here we are only concerned with the outermost, scratched interface.

Our approach is to model scratched materials by combining the reflectance properties of a myriad of individual scratches, exploiting the regularity of their distribution to tackle the high dimensional complexity of the problem. We make the following contributions:

- We introduce a BRDF model for scratches, which, contrary to most previous approaches, explicitly deals with inter-reflections inside a scratch, including Fresnel effects.

- We extend it to a multi-scale spatially-varying BRDF model (SV-BRDF) for the rendering of scratched materials, using arbitrary layered scratch distributions.

- We provide controls at the scratch level: users may adjust their profile, Fresnel coefficients, width, height, density and orientation, while observing results interactively.

- We describe how to evaluate our SV-BRDF in arbitrary settings, either using importance or light sampling.

- We demonstrate our model in progressive rendering for editing and with global illumination for high-quality results.

The main appeal of our approach is to provide a lightweight, fast and controllable solution reproducing the main visual effects of scratched materials, as demonstrated in Figure 1. The accompanying video shows that our method reproduces the characteristic motion of reflections on such materials: strongest highlights are enriched with glint lines that remain coherent in space and time.

2 Previous Work

Appearance modeling is a very wide topic, hence we focus here on approaches that permit to reproduce scratched materials.

2.1 Distribution-based models

Most BRDF models rely on the micro-facet theory [Torrance and Sparrow 1967], where the shape of a BRDF is mainly determined by the normal distribution function (NDF) of micro-facets. Reproducing the appearance of scratched materials requires an anisotropic BRDF, and hence an anisotropic NDF. To this end, Ward [1992] first proposed to use an elliptical Gaussian NDF. Then, more complex anisotropic distributions including an additional Fresnel term have been introduced (e.g., [Ashikhmin and Shirley 2000; Dong et al. 2015]). These analytic models are fast to evaluate but they are valid only for very distant views, assuming that the structure of the micro-geometry is never to be distinguished.

More recent work has attempted to extend distribution-based BRDF models to handle multiple scales, starting from a detailed normal map or height map. A first body of methods [Tan et al. 2005; Tan

et al. 2008; Han et al. 2007] builds a multi-modal NDF from the input normal map. However, the presence of multiple modes involves non-linear interpolation across scales to ensure smooth transitions. A second family of approaches makes use of a unimodal anisotropic NDF which is updated at increasing scales. For instance, Bruneton et al. [2010] use trochoid averaging for the special case of ocean surfaces, whereas Olano et al. [2010] and Dupuy et al. [2013] make use of surface slope moments working for arbitrary but static input displacement maps.

Regarding scratched materials, distribution-based models are limited in two respects. First, they only consider single-scattering and include, at best, approximate shadowing and masking terms. However, we anticipate that inter-reflections inside a scratch impact the visual material appearance. Second, a smooth NDF is valid only when a large number of scratches fall below a pixel. Hence costly sub-pixel sampling is required for intermediate scales, otherwise details will be over-smoothed during transitions.

2.2 Data-driven models

A different approach to the modeling of materials with spatial variations relies on dense and discrete data that can either be measured or computer-generated. The most general approach is the 6D Bidirectional Texture Function (BTF), which depends on two additional spatial coordinates compared to a 4D BRDF. Ma et al. [2005] demonstrate multi-scale renderings of BTFs by means of a Laplacian pyramid. However, owing to their high-dimensionality, BTFs are restricted in terms of spatial and angular resolutions, which is problematic to reproduce the high frequency of scratched materials. Anisotropic BRDF measurements [Filip et al. 2014] already exhibit reduced angular resolutions; measuring high-quality BTFs of scratched materials would be even more difficult.

A number of alternatives to BTFs has been recently proposed. Wu et al. [2009; 2011] introduced two of them, one based on characteristic point maps, and the other one on pre-computed matrices. Both deal with arbitrary micro-geometry. The latter even supports dynamic edition while obtaining renderings for distant views at interactive rates. However, they require costly pre-computations and induce large memory footprints limiting the angular resolution too much to be adapted to scratched materials. Yan et al. [2014] proposed a hierarchical pruning technique to efficiently evaluate the NDF found under each pixel footprint from very high-resolution normal maps. The data-driven NDF is then plugged in a micro-facet model. To the best of our knowledge, this is the first method to produce convincing renderings of scratched materials, exhibiting vivid glint lines. However, this approach ignores inter-reflections inside scratches and requires very high resolution maps to faithfully represent all the variations of the micro-geometry. Very recently, Dong et al. [2015] studied the generation of effective NDFs from measured micro-geometry of metals. Their main contribution is in the prediction of brushed metal BRDFs using data-driven NDFs from profilometer measurements. They also present a compact ellipsoidal NDF model to achieve 2D variations from 1D measured exemplars. This latter approximation smoothes out fine variations and shares the limitations of other distribution-based models (i.e., no inter-reflection, approximate shadowing-masking).

2.3 Primitive-based models

A third approach consists in modeling materials based on distributions of small primitives. Poulin et al. [1990] pioneered the technique by using distributions of micro-cylinders, yielding anisotropic BRDFs. As with other BRDF models, inter-reflections are ignored and micro-geometry is never visible. Individually-visible scratches have been investigated for close-up render-
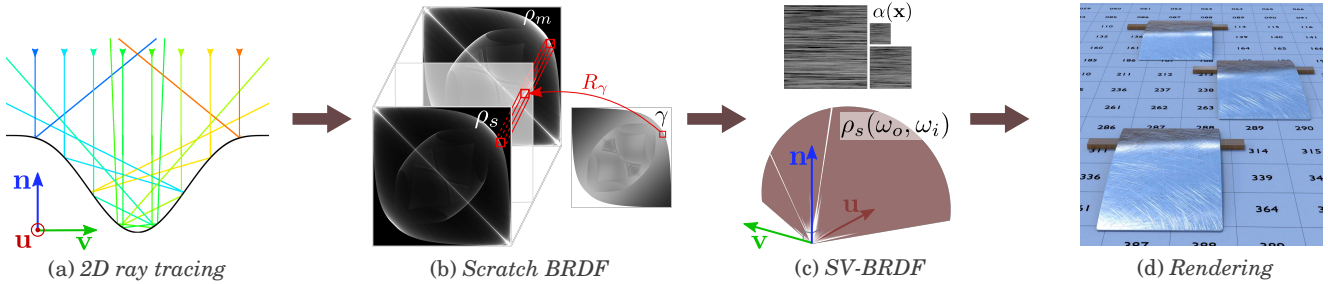
(a) 2D ray tracing   (b) Scratch BRDF   (c) SV-BRDF   (d) Rendering

Figure 3: *(a) The BRDF for a single mirror scratch is simulated using 2D ray tracing (each ray is given a consistent color across reflections for illustration purpose). (b) The full 3D scratch BRDF with Fresnel effects is approximated using an adequate interpolation between a pair of 2D BRDF slices, controlled by an exponent map γ. (c) At render time, a SV-BRDF is constructed by linear combinations of scratch BRDFs (bottom) under a pixel or ray footprint based on the relative area of scratches (top). (d) Our SV-BRDF model is efficiently evaluated at multiple scales, exhibiting both high spatial and angular frequencies.*

ing [Mérillou et al. 2001; Bosch et al. 2004; Bosch and Patow 2010]. However, these methods do not deal with inter-reflections inside scratches, nor do they consider distributions of scratches for rendering in more distant views.

Cloth-rendering has also motivated the development of material models based on fiber primitives. Pont et al. [2003] have first demonstrated how simple sinusoidal fiber profiles lead to complex split off-specular reflections and sophisticated grazing angle effects. However, their analysis is restricted to the plane of incidence and ignores inter-reflections. More recent work [Irawan and Marschner 2012; Sadeghi et al. 2013] provide complete descriptions of BRDF models from fiber properties. They rely on the mirror reflection cone first introduced by Kajiya [1989] and ignore inter-reflections between fibers. Irawan et al. [2012] also provide a texture model, even though multi-scale rendering with transitions between their BRDF and texture models is not demonstrated.

3 Overview

Our approach is *tailored* to the modeling and multi-scale rendering of scratched materials. We introduce a specific solution to deal with the high angular and spatial frequencies typical of such materials, without restricting the choice of scratch profiles or distributions. The different steps of our method are illustrated in Figure 3.

Our BRDF model is both primitive-based and data-driven, as described throughout Section 4. The BRDF for a single scratch is *simulated* using a 2D ray-tracing algorithm (Figure 3a), including inter-reflections and realistic Fresnel effects. The result is stored in the three chanels of a 2D texture (Figure 3b) thanks to a parametrization inspired by the mirror cone of Kajiya [1989].

The SV-BRDF (Figure 3c) is obtained as a *linear combination* of scratch BRDFs (bottom) driven by the area of scratches under a pixel or ray footprint (top). The rendering of our scratched materials at multiple scales (Figure 3d) is covered in Sections 5 and 6, where we detail evaluation methods using either light or importance sampling, in both direct and global illumination settings.

4 Scratched material model

Hypothesis We define scratches as extruded 1D profiles that are *dug into* a base surface, which is considered unaltered in-between scratches. As in previous work, we assume the base surface is *locally flat*, with scratches in a same layer running straight and parallel to each others. In other words, surface curvatures and
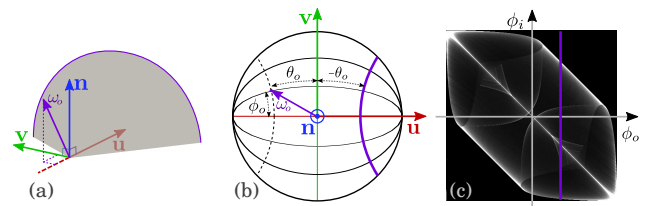

(a)   (b)   (c)

Figure 4: *(a) For a mirror scratch aligned with $\mathbf{u}$, light reflected in the outgoing direction $\boldsymbol{\omega}_o$ lies on a half-cone of directions (in purple). (b) Expressing any direction by a pair of scratch-dependent angular coordinates (θ, ϕ), the half-cone becomes a θ-isocurve. (c) A mirror scratch BRDF is reduced to a 2D function of ϕ angles.*

scratch direction variations (splay and geodesic curvature) are considered negligible at the scratch scale. These conditions ensure that all material-level inter-reflections will be *confined within a single scratch*. We explain in Section 5.3 how to relax our hypothesis to handle scratch profiles exhibiting bulges. We further restrict the micro-BRDF inside a scratch to be a Fresnel reflector. However, the micro-BRDF outside a scratch, which corresponds to the base material, may be chosen freely.

As demonstrated in our results, these assumptions are reasonable for the modeling of scratched materials, enabling a wide range of scratch profiles, distributions and appearances. Most importantly, they allow us to devise a compact and efficient SV-BRDF model as explained in the following.

4.1 Mirror scratch BRDF

Since inter-reflections are confined to a single scratch, each scratch may safely be considered independently of others. We thus start by building a BRDF model for a single *mirror* scratch extruded from an arbitrary profile. As illustrated in Figure 4a, an important property of such an ideal scratch is that light directions contributing to reflection lie on a half-cone around the scratch direction $\mathbf{u}$. This property was first observed by Kajiya [1989] for fibers. The aperture of this cone depends on viewing elevation. Similarly, we identify the family of cones by mapping any unit vector $\boldsymbol{\omega}$ in the hemisphere to angular coordinates $(\theta, \phi) \in [-\frac{\pi}{2}, \frac{\pi}{2}]^2$ given by:

$$(\theta, \phi) = (\sin^{-1}(\boldsymbol{\omega} \cdot \mathbf{u}), \operatorname{atan2}(\boldsymbol{\omega} \cdot \mathbf{v}, \boldsymbol{\omega} \cdot \mathbf{n})),$$

where $\mathbf{n}$ is the surface normal, and $\mathbf{v}$ is a vector orthogonal to both $\mathbf{n}$ and $\mathbf{u}$, hence forming a surface tangent frame.

Let $\rho_m(\theta_o, \phi_o, \theta_i, \phi_i)$ denote our scratch BRDF in this parametrization, where the subscript m stands for *mirror*, and the subscripts o and i refer to the *outgoing* and *incoming* directions respectively. The aforementioned half cone of reflection implies that for a given view direction, only lights on the iso-curve $\theta = -\theta_o$ will contribute (purple curve in Figure 4b). In other words, ρ_m will be zero whenever $\theta_i \neq -\theta_o$, and the dimension of our BRDF can be reduced from 4 to 3. Moreover, since a scratch consists of an extruded profile, varying the view or light elevations along the scratch direction (i.e., θ_o or θ_i) will not change the BRDF behavior along the respective cone. Therefore, ρ_m is invariant with respect to θ, and our BRDF simplifies to a 2D function:

$$\rho_m(\theta_o, \phi_o, \theta_i, \phi_i) = \begin{cases} \rho_m(\phi_o, \phi_i) & \text{if } \theta_i = -\theta_o, \\ 0 & \text{otherwise.} \end{cases} \quad (1)$$

This dimensionality reduction allows us to quickly pre-compute our mirror scratch BRDF model within a compact and discrete $M \times M$ 2D grid parametrized by ϕ_o and ϕ_i (see Figure 4c) that we fill using ray-tracing and density estimation. The unique input of our algorithm is a scratch profile function for which some examples are depicted in Figure 5. For each discrete value of ϕ_i, we shoot $N \gg M$ rays at random positions over a scratch, perform recursive ray-tracing with mirror reflection within the scratch until the ray exits the scratch with a direction ϕ_o. We accumulate its relative integrated quantity of radiation $\frac{M}{N \cos(\phi_o)}$ into the respective grid entry, where $\cos(\phi_o)$ stands for the classic cosine term in our parametrization. Thanks to the above properties with respect to θ, all computations are performed in the 2D plane supporting the scratch profile (see Figure 3a for an illustration), that is for $\theta_i = -\theta_o = 0$.

By construction, our mirror scratch BRDF ensures energy conservation as well as Helmholtz reciprocity up to numerical accuracy. This last property implies a symmetry along the diagonal of the 2D domain. For symmetric scratch profiles, the BRDF also exhibits a symmetry along the anti-diagonal. We exploit these two symmetries to get up to 4 times the number of accumulated rays for free.

Figure 5 depicts the behavior of ρ_m when changing the input scratch profile. We visualize the impact of the number of inter-reflections using red, green and blue colors for the first, second and higher-order bounces respectively. Steeper scratch profiles lead to an increased number of inter-reflections, while flatter profiles tend toward perfect mirror reflection (the anti-diagonal where $\phi_i = -\phi_o$, in red). Measured profiles result in more complex BRDFs; in particular, they are not symmetric across the anti-diagonal.

4.2 Full scratch BRDF

The scratch BRDF of the previous section is an idealized model since no real-world material comes anywhere close to a perfect mirror. We thus now explain how to include Fresnel effects inside a scratch to obtain a more physically-plausible scratch BRDF.

In the micro-facet theory, Fresnel effects are easily included via a single multiplicative term. This is because only single scattering is considered, with additional inter-reflections removed by shadowing. In our case, computing the Fresnel effect is necessarily more involved because intensity along a light path must be attenuated *at each* specular reflection according to Fresnel formula. Since this attenuation depends on the angle θ made between the ray and the micro-surface normal in 3D, a scratch BRDF with Fresnel effects can no longer be represented exactly with a 2D function.

To get a better idea of the influence of the Fresnel effect, we have simulated the full 3D scratch BRDFs with Fresnel attenuation using either the approximation of Schlick [1994] for dielectrics, or the
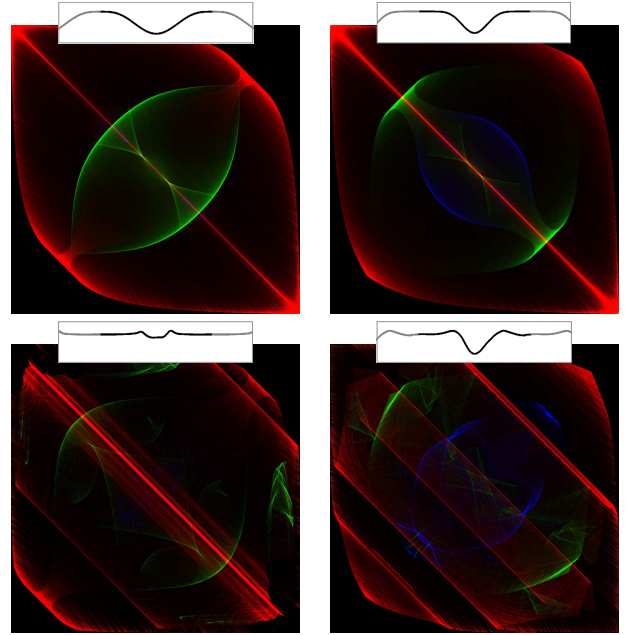

Figure 5: *Scratch profiles (shown on top) and their corresponding mirror scratch BRDFs (1st, 2nd and higher-order bounces in red, green and blue respectively). Top row: a pair of symmetric quartic profiles yield doubly symmetric BRDFs, with the steepest profile (right) yielding more higher-order bounces. Bottom row: a pair of measured profiles yields more complex BRDFs with only one symmetry (due to reciprocity). Even though the two profiles on the right column have a similar global shape, their BRDFs differ significantly. In particular, the peak on the anti-diagonal (i.e., mirror reflection) is absent on the bottom right BRDF due to the lack of flat profile portions.*
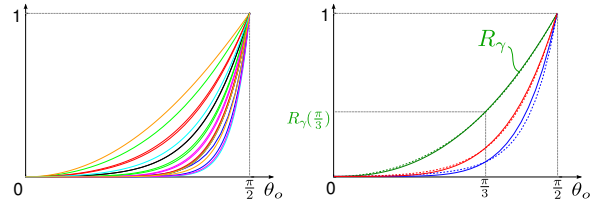

Figure 6: *Left: A subset of reflectance curves showing reflectance as a function of θ_o for fixed (ϕ_o, ϕ_i) pairs, where values have been linearly remapped to the $[0, 1]$ range. Right: Our approximations R_γ (solid lines) for three of these curves (dashed lines).*

approximation of Laznyi et al. [2005] for conductors. In practice, we compute a full scratch BRDF $\rho_s(\theta_o, \phi_o, \phi_i)$, where s simply stands for *scratch*, using our 2D ray-tracer for a range of θ_o values and storing the results in a 3D grid. To investigate variations in the θ_o dimension, we plot reflectance curves corresponding to fixed (ϕ_o, ϕ_i) pairs in Figure 6-left, where we have linearly remapped reflectance values to the $[0, 1]$ range. Here we have used the approximation of Laznyi et al., but Schlick's approximation gives very similar results. As shown in Figure 6-right, the remapped reflectance curves are well approximated by a function of the form $R_\gamma(\theta_o) = \left(\frac{2|\theta_o|}{\pi}\right)^\gamma$. We could have used a non-linear fitting procedure to recover the exponent γ for each reflectance curve; however, we found the following direct fitting solution to be sufficient and much faster. For each (ϕ_o, ϕ_i) pair, we evaluate ρ_s at $\theta_o = \frac{\pi}{3}$, apply the linear remapping with respect to ρ_s at $\theta_o = \{0, \frac{\pi}{2}\}$, and invert $R_\gamma(\frac{\pi}{3})$ to estimate the exponent γ. We chose not to use the exact Fresnel formula in our approach, since the gained accuracy

would have been lost with our angular approximation anyway.

The full scratch BRDF ρ_s with Fresnel effects is then approximated using a simple linear blending of the form:

$$\rho_s(\boldsymbol{\omega}_o, \boldsymbol{\omega}_i) \approx R_\gamma(\theta_o)\, \rho_s(\tfrac{\pi}{2}, \phi_o, \phi_i) + (1 - R_\gamma(\theta_o))\, \rho_s(0, \phi_o, \phi_i), \quad (2)$$

where γ stands for $\gamma(\phi_o, \phi_i)$. This is illustrated in Figure 3b.

It can easily be shown that this equation ensures energy conservation as well as reciprocity (all involved terms remain the same when replacing θ_o by $\theta_i = -\theta_o$). We emphasize that our approximation does not require to simulate the full 3D scratch BRDF; only 3 slices at $\theta_o = \{0, \tfrac{\pi}{3}, \tfrac{\pi}{2}\}$ need to be computed. In addition, since θ_o only has to be considered for the evaluation of Fresnel formula, it does not affect ray propagation, which is thus performed in 2D as in Section 4.1. This avoids numerical issues that would otherwise occur when simulating a BRDF slice at $\theta_o = \tfrac{\pi}{2}$. Finally, since we have $F_\gamma(\tfrac{\pi}{2}) = 1$, the scratch BRDF behaves as a mirror at grazing angles, hence $\rho_s(\tfrac{\pi}{2}, \phi_o, \phi_i) = \rho_m(\phi_o, \phi_i)$. As a result, our full scratch BRDF consists of a triplet of 2D functions: $\rho_s(0, \phi_o, \phi_i)$, $\rho_m(\phi_o, \phi_i)$ and $\gamma(\phi_o, \phi_i)$, which we store in the 3 channels of a single 2D texture.

For a given view direction $\boldsymbol{\omega}_o$, a slice of the scratch BRDF corresponds to a 1D function on the half-cone of reflection, as illustrated in Figure 3c-bottom (white peaks). Observe in particular the split off-specular peaks due to both profile shape and inter-reflections.

Comparisons   Before adding spatial variations to our model, we compare it with two other anisotropic BRDF models in Figure 7 on a simple yet challenging scratch profile: the steep quartic function of Figure 10a. We compute a ground truth image by simulating the full 3D scratch BRDF with Fresnel effects thanks to our dedicated raytracer (Figure 7f) and provide difference images in the bottom right corner for each model. We use a refractive index of $n = 2.91$ and an extinction coefficient of $k = 3.09$ (this matches iron at a wavelength of $650nm$). Figure 7a uses the Ashikhmin-Shirley model [2000], where we have set $n_u = 10^4$ and $n_v = 2$ in order to focus most of the energy into the half cone of reflection, which is appropriate for scratched materials. Any other single-lobe analytic BRDF model would produce similar results. Figure 7b uses a tabulated NDF as in previous works [Yan et al. 2014; Dong et al. 2015]: we adapt our 2D raytracer to only compute the first bounce off the NDF of a scratch profile, and store the result in our 2D parametrization. In both cases, Fresnel and shadowing/masking effects are separately applied as multiplicative terms. Figure 7c demonstrates that our approach, even when limited to a single bounce, is superior to other models (see difference images), which is due to a more accurate treatment of shadowing and masking. When adding multiple bounces, as seen in Figure 7d, our model becomes nearly indistinguishable from the ground truth. The influence of inter-reflections depends on the choice of parameters for the Fresnel effect, which reduces intensity at each successive bounce. This is clearly seen in Figure 7e, where we use our mirror scratch BRDF, hence giving equal contributions to bounces of different orders. Inter-reflections also have a stronger visual impact when using steep instead of shallow scratch profiles, as shown in Figure 14. In any case, inter-reflections come for free in our approach.

4.3   Spatially-varying BRDF

Our full scratch BRDF may already be used to model uniform materials exhibiting densely packed parallel scratches (with no space in-between). As shown in Figure 1a, this reproduces well the directional smearing of reflections observed in Figure 2-right. However, variations in the spacing, width, height and orientation of scratches are necessary to reproduce glint lines and patterned highlights.
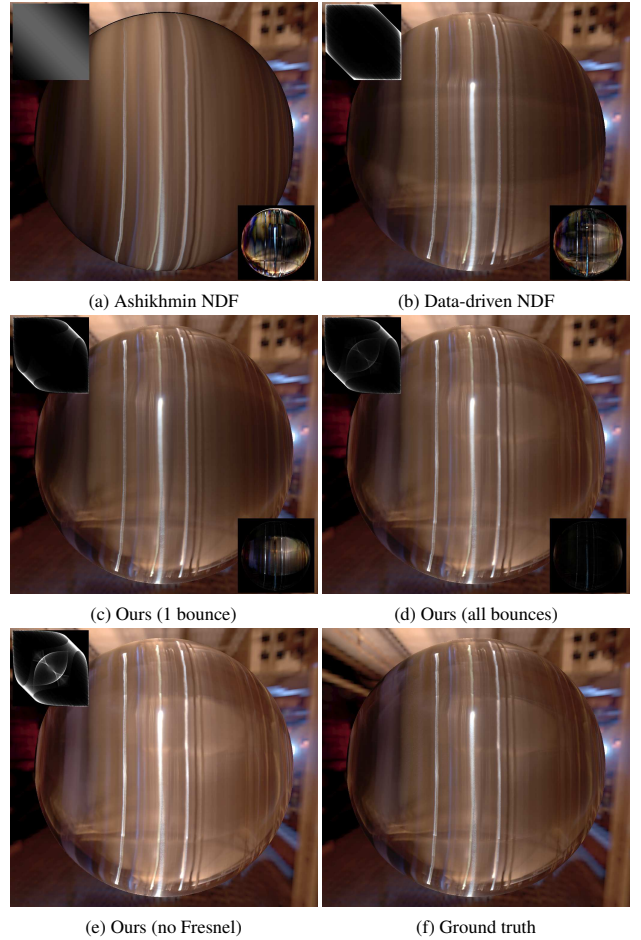

(a) Ashikhmin NDF   (b) Data-driven NDF

(c) Ours (1 bounce)   (d) Ours (all bounces)

(e) Ours (no Fresnel)   (f) Ground truth

Figure 7: *Comparison of anisotropic BRDF models; the top-left insets show a BRDF slice at $\theta_o = 0$ in our parametrization, the bottom right insets show the difference with ground truth (boosted 8 times). (a) Rendering with the Ashikhmin-Shirley BRDF exhibits uniformly smeared reflections. (b) Using a data-driven NDF produces more realistic non-uniform smearing. (c) Our scratch BRDF model without inter-reflections is similar to (b), with more accurate masking and shadowing. (d) Adding inter-reflections produces a result nearly identical to the ground truth in (f). (e) Inter-reflections become strengthened when Fresnel effects are discarded.*

Let us first consider a single scratch layer, which in our approach amounts to fix both the orientation and profile of scratches. An important property of our scratch BRDF is that it is invariant to isotropic transformations of the scratch profile. Hence a same layer may hold a distribution of parallel scratches of different sizes, provided they have the same profile and aspect ratio. Any such distribution of scratches may be used and we describe some of our implementation choices in Section 5.3. For now, it is sufficient to consider that scratches in a layer are localized by an indicator function $\mathbf{1}_S : \mathbb{R}^2 \to \{0, 1\}$. Scratches of different sizes and spacings lead to different indicator functions $\mathbf{1}_S$, which in turn modify the *relative area* α of scratches under a given pixel or ray footprint $\mathcal{P}$:

$$\alpha(\mathbf{x}) = \int_{\mathcal{P}(\mathbf{x})} \mathbf{1}_S(\mathbf{y})\, d\mathbf{y}. \quad (3)$$

While ρ_s describes the angular properties of our model, α characterizes its spatial variations in a multi-scale manner (see Figure 3c-top). This explicit spatial and angular decoupling is central to the

performance of our approach. An implicit assumption is that $\mathbf{1}_S$ agrees with the physical width of scratches. Likewise, the footprint $\mathcal{P}$ should enclose at least one scratch, which defines the minimum scale at which our SV-BRDF model operates.

Up until now we have considered a single scratch layer. To deal with multiple layers, we first compute a combination of scratch BRDFs weighted by relative area using $\bar{\rho}(\mathbf{x}, \boldsymbol{\omega}_o, \boldsymbol{\omega}_i) = \sum \alpha_k(\mathbf{x})\rho_{s,k}(\boldsymbol{\omega}_o, \boldsymbol{\omega}_i)$, where we have added a k subscript to denote the k-th layer. This simple linear combination is not guaranteed to conserve energy though, since at locations where scratches overlap, the combined relative area $\bar{\alpha}(\mathbf{x}) = \sum \alpha_k(\mathbf{x})$ might be greater than 1. In that case, we normalize the accumulated BRDF $\bar{\rho}$ by the sum of weights $\bar{\alpha}$. Otherwise, $1 - \bar{\alpha}$ represents the ratio of scratch-free surface for which we apply the base material ρ_b. Putting it all together, our final SV-BRDF model boils down to:

$$\rho(\mathbf{x}, \boldsymbol{\omega}_o, \boldsymbol{\omega}_i) = \begin{cases} \bar{\rho}/\bar{\alpha}(\mathbf{x}) & \text{if } \bar{\alpha}(\mathbf{x}) > 1 \\ \bar{\rho} + (1 - \bar{\alpha}(\mathbf{x}))\rho_b & \text{otherwise.} \end{cases} \quad (4)$$

This equation avoids the computation of the complex BRDF occuring at scratch intersections, replacing it with a simpler average of scratch reflectances. As shown in Figure 3d, our model is sufficient to reproduce convincing glint lines and patterned highlights at multiple scales. Its simplicity is key to efficient evaluation and sampling in complex scenes as explained next.

5 Rendering with our SV-BRDF model

Our SV-BRDF model is naturally adapted to importance sampling (Section 5.1), but requires special considerations to work with light sampling (Section 5.2). We focus in these two subsections on the radiance reflected by a *single* scratch. Indeed, thanks to the simple formulation of our SV-BRDF model, the total reflected radiance is obtained by a linear combination of the radiances reflected by individual scratches, following Equation 4. Even if the choices of scratch profiles and distributions are arbitrary, we briefly describe our assets in Section 5.3 for completeness.

5.1 Importance sampling

Rendering with our scratch BRDF model amounts to a 1D integral along the half cone of reflections in the local upper hemisphere. We compute the radiance L_r reflected off a single scratch with importance sampling using numerical integration:

$$L_r(\boldsymbol{\omega}_o) = \frac{1}{N} \sum_{i=1}^{N} \frac{\rho_s(\boldsymbol{\omega}_o, \boldsymbol{\omega}_i) cos(\phi_i)}{\text{PDF}_{\phi_o}(\phi_i)} L_i(\boldsymbol{\omega}_i), \quad (5)$$

with N the number of samples and L_i the incoming radiance. The sample directions $\boldsymbol{\omega}_i$ are randomly generated on the reflection cone (i.e., where $\theta_i = -\theta_o$) following a 1D probability distribution function based on the mirror scratch BRDF: $\text{PDF}_{\phi_o}(\phi_i) = \rho_m(\phi_o, \phi_i) cos(\phi_i)$. To this end, we pre-compute the 1D inverse cumulative distribution functions (CDF) for each ϕ_o-slice of the texture storing ρ_m, which is both fast and compact. We could have instead pre-computed and stored the inverse CDF for the full scratch BRDF ρ_s, but this would have been both time and memory consuming. Our approach still remains more accurate than relying on a NDF as done in previous work, since we are taking interreflections into account. Note that the cosine term in Equation 5 cancels out with this choice of PDF. Moreover, when Fresnel effects are discarded, only the $L_i(\boldsymbol{\omega}_i)$ term remains. When rendering with multiple scratch layers, we distribute N samples based on layer importance, which is simply given by α_k in our model.
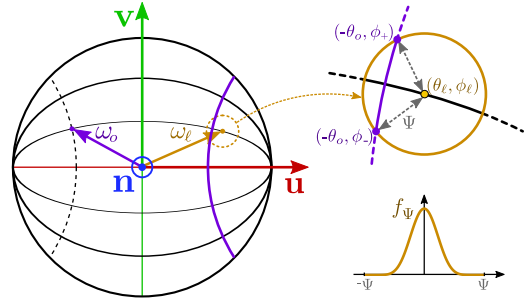

Figure 8: *Geometric intersection between the support of a sample light direction $\boldsymbol{\omega}_\ell$ (in orange) and the half cone of reflection (in purple). The intersected segment (solid purple curve, top right) is identified by its start/end angles $\phi_\pm$. For the falloff function (bottom right), we use $f_\Psi(\psi) = (1 - (\psi/\Psi)^2)^2$.*

5.2 Light sampling

When light samples are provided by the renderer, as with (virtual) point light sources for instance, we are faced with a problem: since sample light directions are infinitely small, they will never exactly lie *on* the half cone of reflection. Following Yan et al.[2014], we assign a narrow reconstruction kernel function around a sample light direction $\boldsymbol{\omega}_\ell$ to prevent this from happening. Any isotropic kernel would do; in practice we use a function $f_\Psi : [-\Psi, \Psi] \rightarrow [0, 1]$ with a compact support of Ψ equivalent to one degree as shown in figure 8. We then locate the direction $\boldsymbol{\omega}'_\ell$ *on* the mirror cone that is closest to $\boldsymbol{\omega}_\ell$: it is given by $(-\theta_o, \phi_\ell)$ in our angular coordinates. We next compute the intersections of the half cone of reflection with the support of f_Ψ (see Figure 8):

$$\phi_\pm = \phi_\ell \pm \sqrt{\Psi^2 - (\theta_\ell - (-\theta_o))^2}.$$

The radiance L_r reflected by a single scratch for a single light sample of intensity L_ℓ is again computed by numerical integration:

$$L_r(\boldsymbol{\omega}_o) = \frac{(\phi_+ - \phi_-)L_\ell}{\pi N} \sum_{i=1}^{N} \rho_s(\boldsymbol{\omega}_o, \boldsymbol{\omega}_i) \cos(\phi_i) f_\Psi(\psi_{i,\ell}), \quad (6)$$

with N the number of samples, $\boldsymbol{\omega}_i$ a direction on the cone segment given by $(-\theta_o, \phi_i)$ in our angular coordinates (with ϕ_i uniformly distributed in the $[\phi_-, \phi_+]$ range), and $\psi_{i,\ell}$ the angle between $\boldsymbol{\omega}_i$ and $\boldsymbol{\omega}_\ell$. We could alternatively use importance sampling of the BRDF on the $[\phi_-, \phi_+]$ range, but we found it not to be necessary considering the small kernel support we use.

When dealing with multiple scratch layers, many combinations of light and scratch directions will make no contribution to reflected radiance. This happens when the half cone of reflections does not intersect the support of f_Ψ, or equivalently when $\Psi < |\theta_\ell + \theta_o|$. It is thus straightforward to prune sample light directions and/or scratch layers *prior* to rendering by simply checking the latter inequality. This is a special case of the method of Yan et al. [2014]. Our solution is simpler only because it is tailored to the geometric properties of scratches, which shows the advantage of having a dedicated scratched material model.

5.3 Implementation details

Scratch profiles We provide users with two different controls over profile shapes, as shown in Figure 5: analytical functions and tabulated height fields from Merillou et al. [2001]. In the latter case, the profile might exhibit bulges that stick out of the base surface, which violates our hypothesis that scratches are only dug inside. We correct this situation by taking the highest profile point as
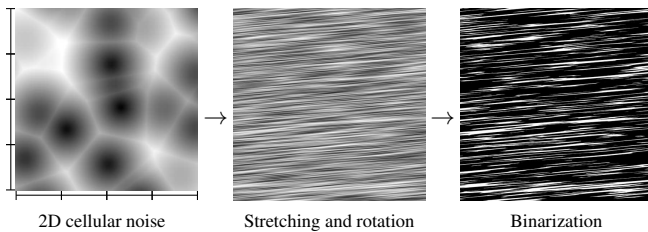
| 2D cellular noise | Stretching and rotation | Binarization |

Figure 9: *A 2D cellular noise is streched and rotated, then binarized to obtain the scratch indicator function. Here we show one layer with $\Theta = 5$, $l = 0.25$, $w = 2 \cdot 10^{-2}$ and $d = 0.5$.*

the base surface height, and making the profile periodic during 2D simulation (ray tracing). This solution works well in practice, especially at grazing angles where profile bulges are naturally masked by nearby, repeated bulges.

Scratch distributions Each layer of a distribution may have its own parametrization on the surface, which may be arbitrarily chosen by the user. In practice, we always rely on a single surface tangent field, which is by default aligned with the u coordinate in texture space. Curved scratches are thus easily achieved by modifying the parametrization as demonstrated in Figure 15. The scratch indicator function $\mathbf{1}_{S_k}$ for a layer k is built by binarizing a noise function stretched along $\mathbf{u}$ (see figure 9):

$$\mathbf{1}_{S_k} = \left\lfloor \mathcal{N} \left(R(\Theta_k) \begin{pmatrix} l & 0 \\ 0 & w \end{pmatrix}^{-1} \begin{pmatrix} u \\ v \end{pmatrix} + \boldsymbol{\xi}_k \right) \right\rfloor_d \qquad (7)$$

where $\mathcal{N}$ is the Worley's cellular 2D noise function [Worley 1996], and $\lfloor x \rfloor_d$ is a thresholding function equal to 1 if $x \le d$ and 0 otherwise. The density parameter d determines the proportion of scratch-versus-base material. Scratch dimensions are controlled by the mean scratch length l along $\mathbf{u}$ and the mean scratch width w along $\mathbf{v}$. Each scratch layer is rotated by an angle Θ_k randomly picked in the user-specified $[-\frac{\Theta}{2}, \frac{\Theta}{2}]$ range. The offset vector $\boldsymbol{\xi}_k \in [-1, 1]^2$ randomly shifts layers with respect to each other. Table 1 collects distribution parameters for all our results, along with the number N of layers.

Image	N	Θ	l	w	d
1 (b)	4	7	0.2	$2 \cdot 10^{-4}$	0.6
1 (c)	16	360	0.1	$1 \cdot 10^{-5}$	0.3
1 (d)	32	360	0.3	$3 \cdot 10^{-4}$	0.05
10 (b,d,e)	6	5	0.1	$5 \cdot 10^{-4}$	0.16
10 (f)	32	360	0.1	$2 \cdot 10^{-4}$	0.05
12 (a)	4	4	0.1	$5 \cdot 10^{-5}$	0.5
12 (b)	16	360	0.1	$2 \cdot 10^{-4}$	0.1
12 (c)	16	360	0.02	$2 \cdot 10^{-5}$	0.1
12 (d)	32	360	0.1	$1 \cdot 10^{-5}$	0.2
14 (a)	4	7	0.02	$7 \cdot 10^{-4}$	0.5
14 (b)	16	360	0.02	$1 \cdot 10^{-5}$	0.2

Table 1: *Distribution parameters used in all our results.*

The computation of the multi-scale relative area α (Equation 3) varies depending on the type of implementation. In a GPU-based renderer, the simplest approach is to build a mipmap texture out of $\mathbf{1}_S$ and rely on graphics hardware to deal with aliasing issues for distant views. In an off-line renderer, the method will depend on the available texture filtering functionalities, which is most often transparent to the user. When working with multiple layers, we rotate and offset the same noise patterns or use different noise functions depending on the desired regularity of the distribution.

6 Results

6.1 Interactive material editing

We have implemented our SV-BRDF model in a custom OpenGL progressive rendering engine. Even though it only works with direct lighting, it provides a sufficiently rich and fast visual feedback for material design. Users may modify the scratch profile, Fresnel parameters, scratch distributions, camera or lighting interactively as shown in Figure 10 and the accompanying video. Performance range from real-time to a few seconds to reach convergence, depending on the number of scratch layers and the type of lighting.

We provide timings obtained with a Nvidia GTX680. Our 2D raytracer is implemented in CUDA to provide fast updates of the scratch BRDF (along with the PDF and inverse CDF) when the profile is modified. Preprocessing takes between 100ms and 300ms for filling a BRDF of resolution 256×256, sending $N = 10K$ samples per ϕ_o slice. Our progressive renderer works in GLSL by accumulating the results of multiple passes, and works with both light sources and environment illumination. With a naive approach, performance scales linearly with the number of scratch layers. We have implemented a practical culling method that discards the k-th layer whenever $\alpha_k < 10^{-6}$, with no perceivable difference in results. We obtain significantly better (sub-linear) timings with this optimization, as summarized in Table 2. Our brushed patterns usually require no more than 6 scratch layers in practice. To achieve fast feedback as demonstrated in the supplemental video, we distribute samples across multiple frames as is typically done in progressive rendering. Compared to the method of Yan et al. [2014] that requires to recompute and sample a NDF for each pixel, our solution is inherently much faster. The reason is that our BRDF amounts to a few texture lookups and simple analytic functions, and supports pre-filtering through the α-maps.

Nb layers	1	6	16	32
Brushed	166	749	1850	3540
Scratched	93	185	419	776

Table 2: *Performance (in ms) for rendering a 615×780 image with 64 samples per pixel and per scratch layer, which provides good visual quality. We use culling based on α as explained in the text and render materials using both brushed and scratched patterns with a varying number of layers. Framed timings correspond to Figure 10a,d,e (for brushed) and Figure 10f (for scratched).*

We have also tested importance sampling on the scratch layers by redistributing the *total* number of samples based on α_k. Figure 11 shows that with only 128 samples for all of the 32 scratch layers already produces clean results in practice compared to uniformly distributing the samples.

The memory footprint of our method is very small: we typically use a 256×256 three-channels texture for the scratch BRDF, a two-channels texture for the precomputed inverse CDF and a one-channel texture for the PDF (1.5 MB without compression). The relative area $\alpha(x)$ is currently stored in a texture, even though it could be computed procedurally if memory usage becomes critical. Note however that the relative area map is usually much smaller than would be a normal map with the same accuracy. More specifically, if the width of the biggest scratch in a high-resolution normal map is w (in texels), then a relative area map with a resolution w times smaller in each dimension will be sufficient. In practice, we observe that a minimum of $w = 10$ texels is necessary to capture a scratch profile, hence our relative area map is at least 100 times smaller than would be a normal map of similar quality.
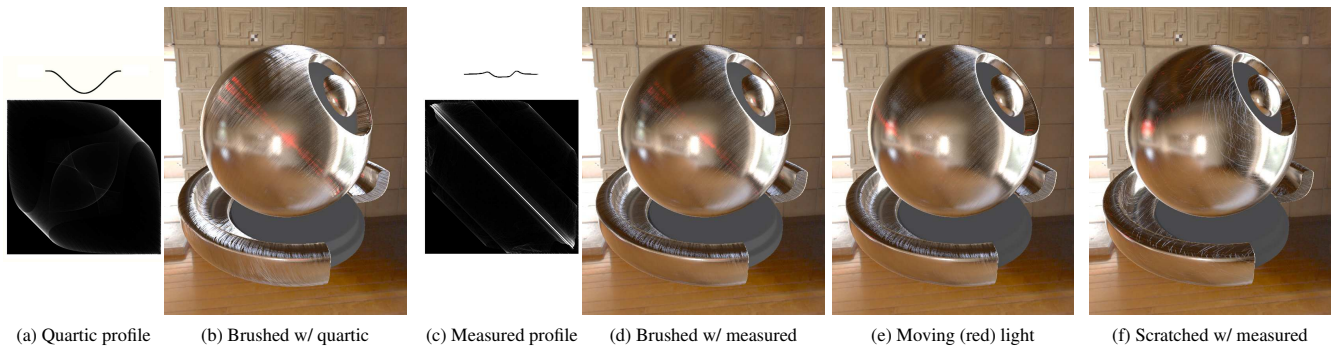
(a) Quartic profile   (b) Brushed w/ quartic   (c) Measured profile   (d) Brushed w/ measured   (e) Moving (red) light   (f) Scratched w/ measured

Figure 10: *Screenshots from an interactive material editing session, at a resolution of 615×780 using 64 samples per pixel and per scratch layer. Progressive rendering is done with direct lighting without shadows to obtain fast feedback. We start in (a) with the BRDF of a quartic scratch profile applied in (b) with a brush pattern (using 6 layers) on our test object lit by an environment map and a red light source. We then switch in (c) to a measured profile [Mérillou et al. 2001], recomputing the BRDF in 100ms. This changes the material appearance in (d): scratches are less pronounced and reflections less elongated. In (e) we rotate the red directional light source, which helps probe the response of the material to different light orientations interactively, taking 749ms. We swicth to a scratch pattern (using 32 layers) in (f), leaving other material parameters unchanged: despite the increase in the number of layers, rendering takes only 776ms thanks to layer culling (see text).*
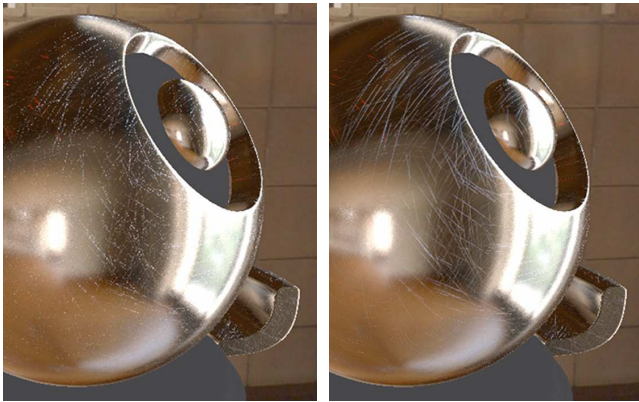

Figure 11: *Rendering with only 128 samples per pixel (for all of the 32 scratch layers) produces noisy scratches. If instead we use importance sampling on scratch layers with the* same *number of samples (right), scratches are much better resolved.*

6.2 Global illumination results

We have implemented our SV-BRDF model as a Blender plugin for the Cycle renderer. This not only permits to demonstrate scratched materials with global illumination, but also allows us to leverage the shading and texturing network functionalities of Cycles to quickly create interesting scratch distributions, as demonstrated in the supplemental video. Figures 1, 12 and 14 include insets that are actual 3D (camera) zooms, *not* image magnifications.

Figure 1a makes use of our scratch BRDF; it corresponds to a densely packed set of parallel scratches. Even in this simple setting, our model already reproduces the characteristic effects of a brushed metal, such as the directionally-smeared reflections observed in Figure 2 right. The main highlight also shows slight variations due to split off-specular peaks in the BRDF. Figure 1b instead uses our full SV-BRDF with a scratch distribution made of 4 layers oriented around a single main direction. This reproduces the appearance of brushed metal, as in Figure 2 middle: note in particular the patterned highlights as well as the mirror-like reflection due to the base BRDF showing up in-between scratches. Figures 1c-d instead use distributions where layers span all orientations, with two

different densities of scratches (with 16 and 32 layers respectively) and deep versus thin profiles. This result is similar to Figure 2 left, where glint lines appear distinctly, especially around strong highlights. Notice also the overall hazy environment reflection obtained when increasing the density of scratches. Inset figures reveal the complex multi-scale behavior of our model: scratches not only get more resolved, but also let different reflection patterns emerge.

Our SV-BRDF is adapted to the modeling of a variety of scratched materials. Figure 12a reproduces the appearance of scratched gold, where scratches are so tiny that they are barely noticeable. Here layers are oriented around a main direction on the surface, which produces an elongated highlight blended with a mirror reflection. Such highlights are also visible in macro-scale inter-reflections, taking an orange tint, as is best seen in the supplemental video. Figure 12b uses two different BRDFs for the scratches and the base surface, imitating the appearance of a scratched painted metal. A sparse set of multi-directional layers reproduces the random deteriorations expected with such materials. Figure 12c makes use of a denser distribution of scratches in all directions, and adds a diffuse term common to the base and scratch BRDFs. Together with an appropriate choice of refractive index ($n = 1.5$), this reproduces the appearance of scratched plastic. Figure 12d uses a similar setting to imitate the appearance of finished wood, by simply using a textured diffuse term in place of a uniform color. Here again, inset figures reveal the behavior of our SV-BRDF model in close-up views, where different reflection patterns emerge compared to zoomed out views.

We have evaluated the performance of our plugin by comparing the anisotropic GGX model of Cycles to our scratch BRDF model (Equation 2). This amounts to consider a single layer of scratches that covers the entire object. Figure 13 shows renderings at 1000 samples per pixel with either BRDF model. Our approach produces a brighter result because of inter-reflections. Compared to the GGX model that takes 484 seconds, our method is slightly slower, taking 550 seconds. However, it should be considered that our implementation remains unoptimized.

We have found that the choice of scratch profiles has a significant impact on material appearance, as demonstrated in Figure 14. Nearly flat profiles (left column) tend to produce distinct off-specular peaks in the BRDF, which show up as multiple highlights on the surface both in densely brushed or sparsely scratched distributions. Steeper profiles (right column) lead to a different material appearance. The number of inter-reflections is increased, hence
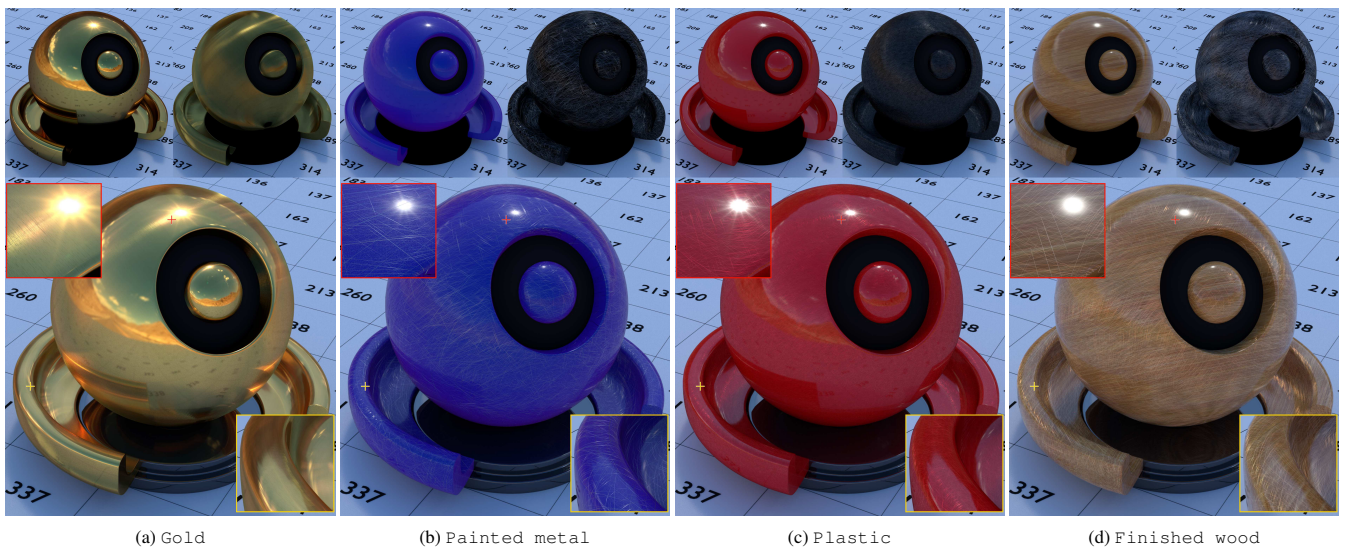
(a) Gold	(b) Painted metal	(c) Plastic	(d) Finished wood

Figure 12: *Four materials rendered with global illumination, using different profile, Fresnel and distribution parameters. Top row: Separate contributions of base and scratch materials. Bottom row: final appearance combining both components. (a)* Gold *(4 layers) is obtained by a regular distribution of sparse tiny scratches that are combined with the base, mirror-like reflection. (b)* Painted metal *(16 layers) makes use of a sparse distribution of scratches in multiple orientations, and a paint BRDF for the base material. (c)* Plastic *(32 layers) uses Fresnel formula for dielectrics and adds a uniformly colored diffuse term to the dense multi-directional scratch layers. (d)* Finished wood *(16 layers) is similar to* Plastic, *except that it uses a sparser set of scratches and a textured diffuse term.*
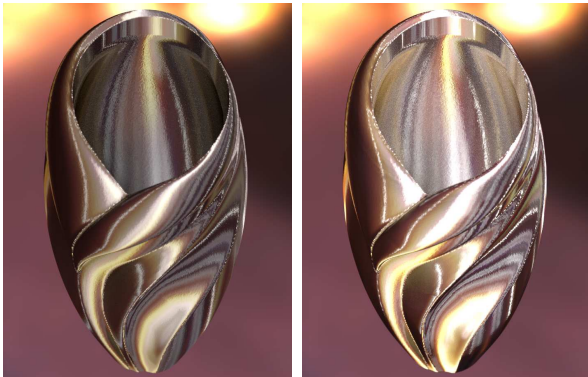

Figure 13: *Global illumination results used for performance comparison. Left: the GGX BRDF of Cycles. Right: our scratch BRDF.*

each additional bounce is attenuated by Fresnel effects and less off-specular peaks occur in the BRDF. As a result, glint lines are more contrasted (see insets) and there are fewer highlights in different locations. Finally, curved scratches are easily achieved by modifying the surface parametrization of each layer, thus enabling more variations without increasing the amount of layers.

7 Limitations and future work

Similar to previous work (e.g., [Dupuy et al. 2013; Jakob et al. 2014b; Dong et al. 2015]), our model makes some simplifications to enable efficient rendering. In particular, we consider idealized and repeated scratches, devoid of imperfections along the extrusion direction. Imperfections might be added by combining techniques dedicated to punctual glitter effects (e.g. [Jakob et al. 2014b]) with our solution. Pitted materials [Pont and Koenderink 2002] could also be modeled and rendered at multiple scales with an approach

similar to ours. Another way to increase visual richness would be to let all scratch parameters (profile, width, height, etc) vary per layer. This would require to devise an analytic scratch BRDF model that could be instanced on the fly, as opposed to running a simulation per scratch profile in preprocess.

In this work we have essentially focused on the SV-BRDF of the outer scratched interface of a material, which has two limitations. First, our spatial blending scheme becomes apparent for extreme close-up views where a scratch is significantly wider than a pixel (see Figure 14-left). This limitation is similar to that of Irawan et al. [2012], whose method also exhibits simplified fiber patterns in close-up views. We plan to extend our SV-BRDF model to a BTF model for extreme close-ups in future work, taking advantage of a scratch structure to avoid storing a full BTF. Second, we have not considered the complex SV-BTDF necessary to model scratched *transparent* materials, such as glass for instance. This is more challenging than reflection due to Snell's law that directly depends on the refractive index, hence making precomputations more difficult.

Ultimately, the decisions made in our work as well as in previous work on scratched and glittery materials should be compared to measured materials. This is a challenge in itself due to the high angular frequency of such BRDFs. Existing work either reduce the angular resolution to capture a full BRDF [Filip et al. 2014] or measure a BRDF for a subset of directions only [Dong et al. 2015]. In the latter case, the authors also measured micro-geometries using an optical profilometer to directly fit their NDF model to brushed metal surfaces. Even though this is limited to single scattering effects (i.e., thin scratch profiles), we believe a similar approach could be used with our model. In our case, the challenge would be to convert the measured micro-geometry to a set of layered scratch primitives, a direction we plan to pursue in future work. We believe that 3D ray-tracing simulations could then be used to compare measured versus fitted micro-surfaces as a validation procedure.
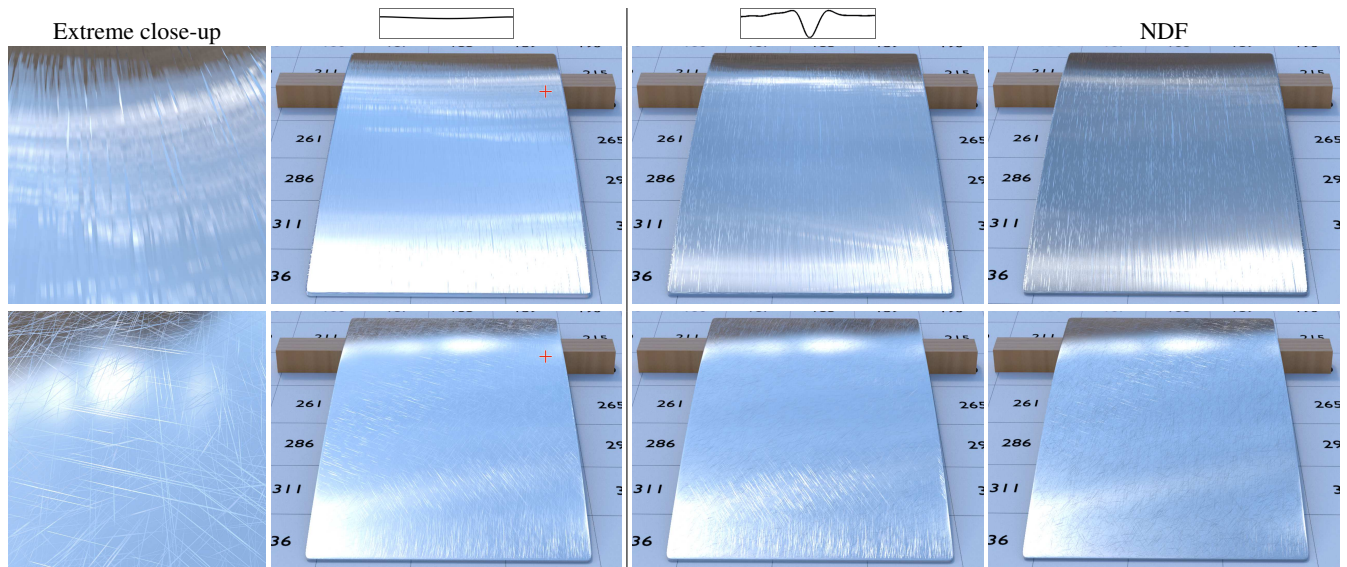
Figure 14: *The choice of scratch profile has a strong impact on material appearance, in either brushed (top row) or scratched (bottom row) patterns. In the first two columns, we use nearly flat profiles, which exhibit multiple highlights due to off-specular peaks in the scratch BRDF. The left-most column shows the limitation of our method for extreme close-up views focused on the area pointed by the red crosses: individual scratches span a few pixels. The third column uses a steeper scratch profile, which yields more constrasted reflections due to multiple bounces being attenuated by Fresnel effects. We compare our results to a NDF-based model in the last column, which shows the importance of inter-reflections. The material appears a lot dimmer when using a NDF since many scratches reflect little or none of the incoming radiance when higher-order bounces are discarded.*
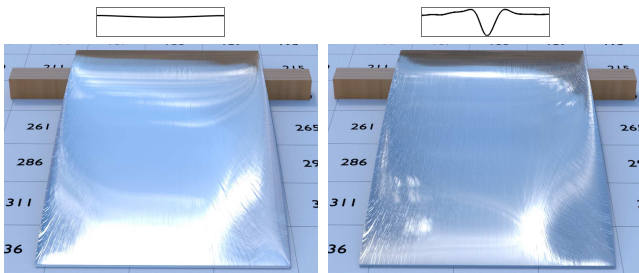

Figure 15: *Curved scratches are obtained by modifying the underlying parametrizations. Here we have used the same profile and distribution as in the middle two columns of Figure 14 (top row).*

8 Conclusion

We have shown that a *dedicated* SV-BRDF model permits to reproduce the high-frequency spatial and angular variations of scratched materials, without requiring heavy computational or memory resources. This has direct practical implications: scratched materials may be designed at interactive rates with fast feedback, and rendered in global illumination with a realistic final look. In particular, our model reproduces the visual appearance of scratched materials, such as the glint lines, patterned highlights and directionally-smeared reflections identified in Figure 2. This is made possible by exploiting the regular micro-structure of scratch distributions, finding a practical balance between precomputations (simulating scratch BRDFs with inter-reflections) and procedural distributions (linearly combining user-defined scratch layers). A benefit of our approach is that it is easily integrated in existing rendering packages and hence extensible to new scratch profiles and distributions. We believe that similar structured SV-BRDF models for other categories of anisotropic materials such as fabrics could also be devised.

References

ASHIKHMIN, M., AND SHIRLEY, P. 2000. An Anisotropic Phong BRDF Model. *Journal of Graphics Tools 5*, 1, 25–32.

BOSCH, C., AND PATOW, G. 2010. Real-time path-based surface detail. *Computers & Graphics 34*, 4 (Aug.), 430–440.

BOSCH, C., PUEYO, X., MERILLOU, S., AND GHAZANFAR-POUR, D. 2004. A Physically Based Model for Rendering Realistic Scratches. *Computer Graphics Forum 23*, 3 (Sept.), 361–370.

BRUNETON, E. 2010. Realtime realistic ocean lighting using seamless transitions from geometry to brdf. *Computer Graphics Forum 29*, 2.

DONG, Z., WALTER, B., MARSCHNER, S., AND GREENBERG, D. P. 2015. Predicting appearance from measured microgeometry of metal surfaces. *ACM Trans. Graph. 35*, 1, 9:1–9:13.

DUPUY, J., HEITZ, E., IEHL, J., AND POULIN, P. 2013. Linear efficient antialiased displacement and reflectance mapping. *ACM Transactions on . . . 6*, 211, 1–11.

FILIP, J., VAVRA, R., AND HAVLICEK, M. 2014. Effective acquisition of dense anisotropic brdf. In *Proceedings of the 22th*

International Conference on Pattern Recognition, ICPR 2014, 2047–2052.

HAN, C., SUN, B., RAMAMOORTHI, R., AND GRINSPUN, E. 2007. Frequency domain normal map filtering. *ACM Trans. Graph. 26*, 3 (July).

IRAWAN, P., AND MARSCHNER, S. 2012. Specular reflection from woven cloth. *ACM Transactions on Graphics 31*, 1 (Jan.), 1–20.

JAKOB, W., D'EON, E., JAKOB, O., AND MARSCHNER, S. 2014. A comprehensive framework for rendering layered materials. *ACM Transactions on Graphics (Proceedings of SIGGRAPH 2014) 33*, 4.

JAKOB, W., HAŠAN, M., YAN, L.-Q., LAWRENCE, J., RAMAMOORTHI, R., AND MARSCHNER, S. 2014. Discrete stochastic microfacet models. *ACM Transactions on Graphics (Proceedings of SIGGRAPH 2014) 33*, 4.

KAJIYA, J. T., AND KAY, T. L. 1989. Rendering fur with three dimensional textures. *SIGGRAPH Comput. Graph. 23*, 3 (July), 271–280.

LAZANYI, I., AND SZIRMAY-KALOS, L. 2005. Fresnel Term Approximations for Metals. In *WSCG'2005 - 13-th International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision'2005*, 77–80.

MA, W.-C., CHAO, S.-H., TSENG, Y.-T., CHUANG, Y.-Y., CHANG, C.-F., CHEN, B.-Y., AND OUHYOUNG, M. 2005. Level-of-detail representation of bidirectional texture functions for real-time rendering. In *Proceedings of the 2005 Symposium on Interactive 3D Graphics and Games*, ACM, New York, NY, USA, I3D '05, 187–194.

MÉRILLOU, S., DISCHLER, J.-M., AND GHAZANFARPOUR, D. 2001. Surface scratches: measuring, modeling and rendering. *The Visual Computer 17*, 1, 30–45.

OLANO, M., AND BAKER, D. 2010. Lean mapping. In *Proceedings of the 2010 ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, ACM, New York, NY, USA, I3D '10, 181–188.

PONT, S. C., AND KOENDERINK, J. J. 2002. Bidirectional reflectance distribution function of specular surfaces with hemispherical pits. *J. Opt. Soc. Am. A 19*, 12 (Dec), 2456–2466.

PONT, S. C., AND KOENDERINK, J. J. 2003. Split off-specular reflection and surface scattering from woven materials. *Appl. Opt. 42*, 8 (Mar), 1526–1533.

POULIN, P., AND FOURNIER, A. 1990. A model for anisotropic reflection. *SIGGRAPH Comput. Graph. 24*, 4 (Sept.), 273–282.

SADEGHI, I., BISKER, O., DE DEKEN, J., AND JENSEN, H. W. 2013. A practical microcylinder appearance model for cloth rendering. *ACM Trans. Graph. 32*, 2 (Apr.), 14:1–14:12.

SCHLICK, C. 1994. An inexpensive brdf model for physically-based rendering. *Computer Graphics Forum 13*, 233–246.

TAN, P., LIN, S., QUAN, L., GUO, B., AND SHUM, H.-Y. 2005. Multiresolution reflectance filtering. In *Proceedings of the Sixteenth Eurographics Conference on Rendering Techniques*, Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, EGSR '05, 111–116.

TAN, P., LIN, S., QUAN, L., GUO, B., AND SHUM, H. 2008. Filtering and rendering of resolution-dependent reflectance models. *IEEE Transactions on Visualization and Computer Graphics 14*, 2 (Mar.), 412–425.

TORRANCE, K. E., AND SPARROW, E. M. 1967. Theory for off-specular reflection from roughened surfaces. *J. Opt. Soc. Am. 57*, 9 (Sep), 1105–1112.

WARD, G. 1992. Measuring and modeling anisotropic reflection. *ACM SIGGRAPH Computer Graphics 26*, 2, 265–272.

WORLEY, S. 1996. A cellular texture basis function. In *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '96, 291–294.

WU, H., DORSEY, J., AND RUSHMEIER, H. 2009. Characteristic point maps. *Computer Graphics Forum 28*, 4, 1227–1236.

WU, H., DORSEY, J., AND RUSHMEIER, H. 2011. Physically-based interactive bi-scale material design. *ACM Transactions on Graphics (TOG)*.

YAN, L.-Q., HAŠAN, M., JAKOB, W., LAWRENCE, J., MARSCHNER, S., AND RAMAMOORTHI, R. 2014. Rendering glints on high-resolution normal-mapped specular surfaces. *ACM Trans. Graph. 33*, 4 (July), 116:1–116:9.