

Optimizing BRDF Orientations for the Manipulation of Anisotropic Highlights

Boris Raymond, Gaël Guennebaud, Pascal Barla, Romain Pacanowski, Xavier Granier

Inria - Bordeaux University - IOGS - CNRS

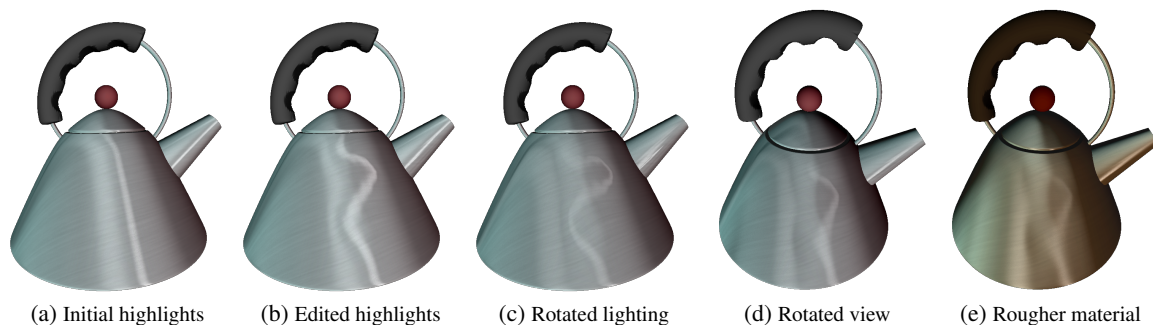


Figure 1: (a) Rendering an object using an anisotropic BRDF [AS00] produces stretched highlights. (b) Our approach lets users directly modify highlight shape by inferring appropriate BRDF orientations. (c-d) The edited highlights retain their deformed shape for nearby lighting or viewing configurations. (e) Highlight shape is still preserved when a rougher material is used.

Abstract

This paper introduces a system for the direct editing of highlights produced by anisotropic BRDFs, which we call *anisotropic highlights*. We first provide a comprehensive analysis of the link between the direction of anisotropy and the shape of highlight curves for arbitrary object surfaces. The gained insights provide the required ingredients to infer BRDF orientations from a prescribed highlight tangent field. This amounts to a non-linear optimization problem, which is solved at interactive framerates during manipulation. Taking inspiration from sculpting software, we provide tools that give the impression of manipulating highlight curves while actually modifying their tangents. Our solver produces desired highlight shapes for a host of lighting environments and anisotropic BRDFs.

Categories and Subject Descriptors (according to ACM CCS): I.3.3 [Computer Graphics]: Picture/Image Generation—Display algorithms I.3.4 [Computer Graphics]: Graphics Utilities—Paint systems

1. Introduction

Objects in our everyday surroundings span a wide range of materials, yet we are able to reliably estimate their properties only by looking at them [Ade01]. Surface highlights are amongst the most informative sources of material appearance: their shape, contrast and distinctness constitute important cues to visual experience [FDA03]. *Highlight shape* in particular is most often determined by surface shape and environment lighting. However, highlights may also come in extended, arched or looped shapes. These are due to microscopic properties of the material, which present elongations across the surface examples abound in nature (e.g., coats, furs and hairs) and are prized in man-made objects (e.g., brushed metals, silk or satin).

In Computer Graphics, materials are often modeled with Bi-directional Reflectance Distribution Functions (BRDF) [Nic77], which are 4D functions of light and view directions. Many materials can be considered isotropic, in which case the BRDF reduces to a 3D function and the orientation of the tangent frame becomes irrelevant. We are instead interested in anisotropic BRDFs since they are responsible for the occurrence of complex highlights, which we term *anisotropic highlights* in the following. By construction, the shape of anisotropic highlights is determined by BRDF orientations across the surface; but it also depends on object shape, lighting and viewing configurations. As a result, it is extremely difficult to predict the appearance of anisotropic highlights on arbitrary-shaped objects.

In this paper, we aim at providing tools that directly manipulate the shape of anisotropic highlights without modifying their other aspects. In terms of BRDF, our goal is to modify local orientations while keeping intrinsic parameters unchanged, which amounts to inferring a *direction field* across the surface. This is a difficult task that has never been attempted as far as we know. We think of our approach as an extension to 3D sculpting where we indirectly “sculpt” a direction field through specified highlight shape. This is also a first step toward the fabrication of objects exhibiting controlled anisotropic material appearance.

This work makes three specific contributions:

- We extend the analytic work of Lu et al. [LKK00] to characterize local tangents of anisotropic highlight curves for arbitrary-shaped objects (Section 3);
- We describe a mesh-based non-linear optimization that infers a direction field from specified anisotropic highlight tangents (Section 4);
- We propose a set of tools for manipulating highlights directly on top of the surface (Section 5).

In practice, we start from a mesh with an initial direction field, which is updated to match user-specified highlight deformations with respect to a reference light direction. Our approach works for a host of lighting environments and anisotropic BRDFs, which we demonstrate in Section 6.

2. Related Work

The manipulation of appearance is a quite recent field of research in Computer Graphics. The BRDF-shop system [CPK06] lets users control the shape of a highlight produced by a distant light source, from which BRDF properties are inferred. Ben-Artzi et al. use pre-computed representations of lighting that permit to edit a separable BRDF model in complex lighting environments [BAOR06, BAERD08]. Another approach to manipulate appearance consists in modifying the environment lighting. The Illumination Brush [OMSI07] lets users paint desired colors on an object with a known BRDF, and then infers diffuse and specular environment maps. The EnvyLight system [Pel10] provides for more global controls and modifies an initial environment through user-provided annotations in the image. Ritschel et al. [ROTS09] provide a solution to edit directly mirror reflections. However, their method is not physically plausible and does not deal with anisotropic materials. Our approach is complementary to these as it addresses a different problem: *orienting* BRDF tangent frames across an object surface to control highlight shape.

Related to our work is the design of smooth direction fields on discrete surfaces [ZMT06, FS07, CDS10]. Given specified field singularities and optional sparse direction constraints, these methods aim at inferring an as smooth as possible direction field. More recently, Knöppel et al. [KCPS13] also showed that singularities can be optimized with a convex formulation. These techniques provide fundamental tools for the manipulation of direction fields; this is not directly usable in our context though, as shown in the supplemental video. Indeed, our application involves

a *complex indirection* between the field that has to be optimized (i.e., BRDF orientations), and the vectors onto which the constraints and desired properties apply (i.e., highlight curve tangents).

The work of Gingold et al. [GZ08] exhibits such an indirection as well, even though in their case the goal is to find the object shape that best matches a specified smooth shading. In their system, the user paints a desired shading onto the object surface, which is translated into a set of constraints for deformation. We build on this interaction metaphor, except that in our case users manipulate anisotropic highlights instead of smooth shading, which modifies local tangent frames instead of surface shape. The two problems require different solutions, as made clear in the next section where we expose the specificities of anisotropic highlights.

3. The shape of anisotropic highlights

We first explain anisotropic BRDFs in terms of micro-facet theory in Section 3.1. Then we conduct a differential analysis whose purpose is to express anisotropic highlight tangents in terms of local surface properties in Section 3.2.

3.1. Micro-facet theory

The traditional approach to model anisotropic BRDFs is through micro-facet theory [TS67]. The surface is then considered to be made of a distribution of microscopic specular facets (perfect mirrors). Different BRDFs are obtained by varying distributions: for most existing models micro-facets are oriented around the surface normal \mathbf{n} . To obtain anisotropic BRDFs, the distribution must be anisotropic as well. A typical choice is to consider a distribution elongated along a tangential direction \mathbf{u} . Although these choices correspond to many materials, some instances such as velvet require different types of distributions [APS00] and are not considered in our study.

Micro-facet theory is best understood by identifying situations where the BRDF exhibits a maximal response (i.e., produces a highlight). As a perfect mirror, a single micro-facet reflects light only when its normal is the bisector between the light ω_{in} and the view ω_{out} , also known as the halfway vector and given by $\mathbf{h} = (\omega_{in} + \omega_{out}) / \|\omega_{in} + \omega_{out}\|$. For an elongated distribution, most micro-facet normals are orthogonal to \mathbf{u} . Therefore, for a given elevation (i.e., $\mathbf{h}^T \mathbf{n} = \text{const}$), reflectance is maximized when $\mathbf{h}^T \mathbf{u} = 0$. This is illustrated

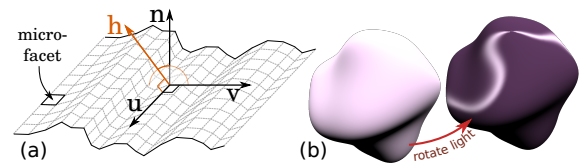


Figure 2: (a) An idealized elongated micro-facet distribution: highlights occur only when \mathbf{h} spans the plane perpendicular to \mathbf{u} . (b) Orienting \mathbf{u} perpendicularly to \mathbf{h} at every surface point yields a strong quasi-uniform reflectance that quickly reverts to a curve when the light is rotated.

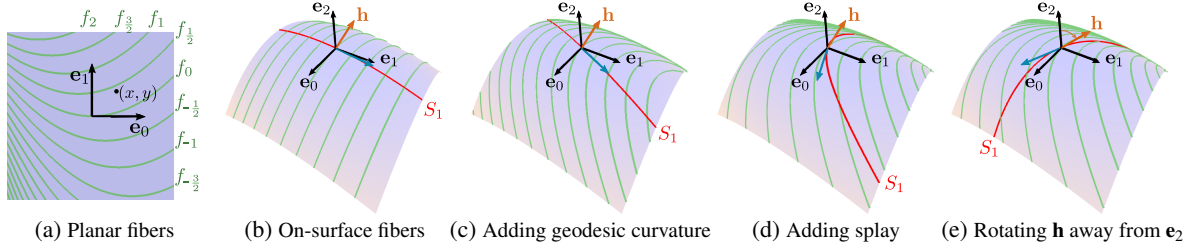


Figure 3: (a) A subset (in green) of a family of fibers ($a = 1.4$, $b = 0.8$) on a plane. (b) Straight fibers applied to a 3D surface ($c_{20} = -1.0$, $c_{11} = 0.1$, $c_{02} = -0.3$) produce a highlight curve (in red) with its tangent direction (in cyan) nearly orthogonal to fibers. (c-d) When fibers bend ($a = 1.4$), then diverge ($b = 0.8$), the highlight curve rotates significantly and its tangent nearly aligns with fibers. (e) Varying $\theta_{\mathbf{h}}$ from 0.5 to 0.8 radians rotates even further the specular curve in the tangent plane.

in Figure 2a where *all* micro-facets are orthogonal to \mathbf{u} ; in this extreme case, reflectance is non-zero only when \mathbf{h} lies in the plane spanned by \mathbf{n} and the vector $\mathbf{v} = \mathbf{n} \times \mathbf{u}$.

Such an observation already suggests a way to manipulate anisotropic highlights: find the direction \mathbf{u}^* that maximizes reflectance at every surface point. We know that $\mathbf{h}^T \mathbf{u}^*$ must equal 0; but we also know that \mathbf{u}^* must be tangent to the surface. Combining both constraints, we obtain two solutions: $\mathbf{u}^* = \pm \mathbf{h} \times \mathbf{n}$. One of these is illustrated in Figure 2b, before and after a small rotation that shows how much anisotropic highlights are sensitive to viewing conditions in this configuration. This is because generic anisotropic highlights take the shape of *curves*, whereas highlight regions correspond to degenerate configurations. We study highlight curves in more detail in the following section.

3.2. Differential analysis

Local specular fibers. In an effort to gain understanding on the shape of anisotropic highlights, Lu et al. have conducted a differential analysis in a simplified case [LKK99, LKK00]. They consider micro-facets to be distributed on tangential specular fibers, hairs or grooves aligned at the origin with $\mathbf{e}_0 = (1, 0, 0)^T$. They use second-order approximations for fiber shape and surface height. They consider constant lighting and viewing directions, and let \mathbf{h} span the plane orthogonal to \mathbf{e}_0 (i.e., $\mathbf{h} = (0, \sin \theta_{\mathbf{h}}, \cos \theta_{\mathbf{h}})^T$). Together, these design choices simplify analysis since they result in a single highlight curve that goes through the origin and for which an analytical form can be found. We quickly recall their approach using their notation (see Figure 3).

Fiber shape $f_y : \mathbb{R} \rightarrow \mathbb{R}^2$ is approximated by:

$$f_y(x) := \left[x, y + \frac{ax^2}{2} + bxy \right]^T,$$

where a and b designate geodesic curvature and splay in the tangent plane. By varying y , one sweeps through the set of fibers, as shown in Figure 3a. There is no y^2 term as fiber spacing is ignored. Surface height is approximated by:

$$z(x, y) := \frac{c_{20}x^2 + 2c_{11}xy + c_{02}y^2}{2},$$

where c_{20} , c_{11} and c_{02} are 2nd-order coefficients that define the curvature of the surface.

Lu et al. derive two formulas for the resulting highlight curve $S_{1,2} : \mathbb{R} \rightarrow \mathbb{R}^3$, using different parameterizations. As detailed in [LKK00], both lead to the same tangent *direction* at the origin. For instance, using x as a parameter:

$$\frac{dS_1(x)}{dx} \Big|_{x=0} = \left[1, -\frac{a \sin \theta_{\mathbf{h}} + c_{20} \cos \theta_{\mathbf{h}}}{b \sin \theta_{\mathbf{h}} + c_{11} \cos \theta_{\mathbf{h}}}, 0 \right]^T. \quad (1)$$

As illustrated in Figure 3b-e, the highlight curve and its tangent direction vary significantly with surface curvature, fibers shape and half-way vector elevation. Lu et al. provide a study of these highlight curves for simple geometric primitives, as well as a detailed account of degenerate cases, but they do not provide any solution for arbitrary-shaped objects.

Extension to arbitrary surfaces. In our approach, the object surface is described by a 2D manifold \mathcal{M} . BRDF orientations are controlled by a direction field $\mathbf{u} : \mathcal{M} \rightarrow T\mathcal{M}$ where $T\mathcal{M}$ denotes the tangent bundle of \mathcal{M} . In this general setting, Equation 1 cannot be used directly to determine highlight curve tangents for three reasons.

First and foremost, the direction of fibers is constrained at the origin to \mathbf{e}_0 . This corresponds in our case to fixing \mathbf{u} , which is precisely the parameter we want to vary. In the following, we thus re-express Equation 1 and its constituents as functions of \mathbf{u} . Our aim is then to write the highlight curve tangent as a world-space vector $\ell(\mathbf{u}) \in T\mathcal{M}$.

Second, we do not model anisotropic materials through fibers but with general elongated micro-facet distributions. However, individual fibers may be seen as streamlines of the direction field \mathbf{u} . Geodesic curvature and splay are then equivalent to the signed curl magnitude and divergence of \mathbf{u} :

$$a = |\nabla \times \mathbf{u}| \quad b = \nabla \cdot \mathbf{u}, \quad (2)$$

where we use $|\nabla \times \mathbf{u}| := (\nabla \times \mathbf{u})^T \mathbf{n}$ as a short notation.

Third, surface shape is given as a quadratic height-field defined with respect to fibers direction. From elemental differential geometry [dC76], this is equivalent to compute curvature κ and torsion τ in the direction of \mathbf{u} :

$$c_{20} = \kappa(\mathbf{u}) \quad c_{11} = \tau(\mathbf{u}). \quad (3)$$

Putting it all together, we rewrite Equation 1 as a *non-oriented* highlight tangent field $\ell : T\mathcal{M} \rightarrow T\mathcal{M}$. The am-

T, \mathcal{M}	Tangent bundle of surface \mathcal{M}
$[\mathbf{t}, \mathbf{b}, \mathbf{n}]$	Surface tangent frame field
τ, κ	Surface torsion & curvature
$\omega_{in}, \omega_{out}$	Lighting & viewing directions
\mathbf{h}	Half vector between ω_{in} & ω_{out}
\mathbf{u}	BRDF direction field
\mathbf{v}	Direction field orthogonal to \mathbf{u}
φ	Angular representation of \mathbf{u}
ℓ	Highlight tangent field

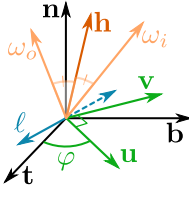


Figure 4: List of notations used in our approach.

biguity in orientation comes from the fact that a highlight curve has no intrinsic parametrization. We first multiply Equation 1 by the denominator of the y coordinate to avoid degenerate cases. Then, using Equations 2 and 3 and replacing (co)sines with dot products, we obtain:

$$\ell(\mathbf{u}) = \pm \mathbf{U} \begin{bmatrix} (-\nabla \cdot \mathbf{u}) \mathbf{h}^T \mathbf{v} - \tau(\mathbf{u}) \mathbf{h}^T \mathbf{n} \\ |\nabla \times \mathbf{u}| \mathbf{h}^T \mathbf{v} + \kappa(\mathbf{u}) \mathbf{h}^T \mathbf{n} \\ 0 \end{bmatrix},$$

where $\mathbf{U} = [\mathbf{u} \ \mathbf{v} \ \mathbf{n}]$ is a local-to-world space transform.

Written in matrix form, this yields:

$$\ell(\mathbf{u}) = \pm \mathbf{U} \begin{bmatrix} 0 & -\nabla \cdot \mathbf{u} & -\tau(\mathbf{u}) \\ 0 & |\nabla \times \mathbf{u}| & \kappa(\mathbf{u}) \\ 0 & 0 & 0 \end{bmatrix} \mathbf{U}^T \mathbf{h}. \quad (4)$$

Note that the resulting highlight tangent is *not* a unit vector. It permits to characterize the degenerate case of Figure 2b, which occurs when $\ell(\mathbf{u}) = 0$. Our own notations are summarized in Figure 4 and used in the remainder of the paper.

Potential highlight curves. An implication of Equation 4 is that it characterizes *potential* highlight curve tangents. Indeed, the first column of the central matrix acts as a projection of the halfway vector \mathbf{h} onto the plane orthogonal to \mathbf{u} . This is thus equivalent (up to a multiplicative factor) to the highlight tangent that would be obtained for a nearby lighting direction. An important consequence is that highlight tangents will remain coherent for small changes of lighting or viewing direction, as shown in Figure 5a-c using Line Integral Convolution (LIC) [CL93]. Hence Equation 4 suggests that, by varying \mathbf{u} , one may not only deviate the trajectory of a single reference highlight, but also of *potential* highlight curves around it. This is a fundamental property on which we ground our editing approach.

Potential highlights exhibit a critical case when $\mathbf{h} = \mathbf{n}$ though. One highlight curve is indeed constrained to pass through the surface point since in this case, $\mathbf{h}^T \mathbf{u} = 0 \ \forall \mathbf{u}$. Such critical points have a simple interpretation, as they correspond to peaks of corresponding *isotropic* highlights. Consequently, varying \mathbf{u} has no effect on the location of critical points as shown in Figure 5d-e. According to Equation 4, it can only modify highlight tangents in a way that exclusively depends on directional surface curvature and torsion.

4. BRDF orientations from specified highlight tangents

Our goal is now to find a BRDF direction field \mathbf{u} such that the potential highlight tangents $\ell(\mathbf{u})$ given by equation 4 match user-specified targets ℓ^* .

Methods for specifying these are deferred to Section 5. We formulate this problem as a minimization of the following least-square energy:

$$E_{\ell}(\mathbf{u}) = \int_{\mathcal{M}} \|\mathbf{L}(\mathbf{u}) - \mathbf{L}^*\|^2 dA, \quad (5)$$

where we classically represent non-oriented highlight tangents by normalized tensors of the form $\mathbf{L} = \ell \ell^T / \|\ell\|^2$. Together, Equations 4 and 5 constitute a highly non-linear optimization problem. Another difficulty that we tackle first is that we must solve for \mathbf{u} in the set of unit tangent vectors.

4.1. Angular representation

Let $[\mathbf{t}, \mathbf{b}, \mathbf{n}]$ denote an arbitrary tangent frame field. We represent the direction field \mathbf{u} as a scalar field $\varphi : \mathcal{M} \rightarrow \mathbb{R}$, which defines \mathbf{u} as a rotation of \mathbf{t} by φ radians around \mathbf{n} . In the $[\mathbf{t}, \mathbf{b}]$ plane, this is equivalent to a 2D rotation $[\bar{\mathbf{u}} \ \bar{\mathbf{v}}]$ where $\bar{\mathbf{u}}$ and $\bar{\mathbf{v}}$ are 2D versions of \mathbf{u} and \mathbf{v} respectively:

$$\bar{\mathbf{u}}(\varphi) = [\cos \varphi \ \sin \varphi]^T \quad \bar{\mathbf{v}}(\varphi) = [-\sin \varphi \ \cos \varphi]^T. \quad (6)$$

As detailed in the appendix, Equation 4 can then be written as a function of φ , which simplifies it to:

$$\ell(\varphi) = \pm [\mathbf{b} \ -\mathbf{t}] \left[\nabla \varphi \bar{\mathbf{v}}(\varphi)^T \ \mathbf{W} \bar{\mathbf{u}}(\varphi) \right] [\mathbf{t} \ \mathbf{b} \ \mathbf{n}]^T \mathbf{h}, \quad (7)$$

where \mathbf{W} is the 2×2 Weingarten map, and $\nabla \varphi$ is the 2D gradient vector of covariant derivatives.

4.2. Discrete setting

We consider the surface \mathcal{M} to be given as a mesh with vertex positions \mathbf{p}_i , normals \mathbf{n}_i and arbitrary unit tangent basis vectors \mathbf{t}_i and \mathbf{b}_i . Our optimization problem then becomes:

$$\varphi^* = \arg \min_{\varphi} \|\mathbf{E}(\varphi)\|^2, \quad E_i(\varphi) = A_i^{\frac{1}{2}} \|\mathbf{L}_i(\varphi) - \mathbf{L}_i^*\|^2, \quad (8)$$

where A_i is the area of the Voronoï cell around vertex i , and $\mathbf{L}_i(\varphi)$ is its potential highlight tangent given in tensor form with respect to φ (via Equation 7). We must now compute the gradient $\nabla \varphi_i$ and Weingarten map \mathbf{W}_i for each vertex i .

Starting with $\nabla \varphi_i$, we consider the first-order approximation

$\hat{\mathbf{u}}_i(\mathbf{x}) = \bar{\mathbf{u}}_i + \mathbf{J}_i \mathbf{x}$ of the vector field $\bar{\mathbf{u}}$ around \mathbf{p}_i expressed in $[\mathbf{t}_i, \mathbf{b}_i]$. Using equation 6, the Jacobian is re-written as $\mathbf{J}_i = \bar{\mathbf{v}}_i \nabla \varphi_i^T$, $(\hat{\varphi}_i(\mathbf{x}) - \varphi_i)$ which explicitly shows that all 1st-order variations of $\bar{\mathbf{u}}$ around \mathbf{p}_i happen along $\bar{\mathbf{v}}_i$. If we project our approximation onto $\bar{\mathbf{v}}_i$, we obtain $\bar{\mathbf{v}}_i^T \hat{\mathbf{u}}_i(\mathbf{x}) = \nabla \varphi_i^T \mathbf{x}$. Using a polar representation for $\hat{\mathbf{u}}_i$ as illustrated in the inset Figure, we get:

$$\|\hat{\mathbf{u}}_i(\mathbf{x})\| \sin(\hat{\varphi}_i(\mathbf{x}) - \varphi_i) = \nabla \varphi_i^T \mathbf{x}. \quad (9)$$

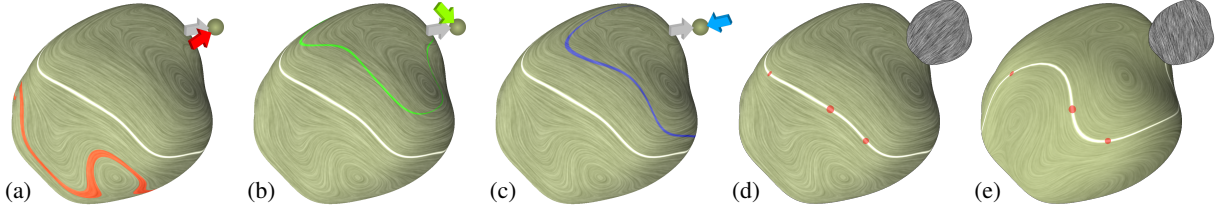


Figure 5: (a-c) The highlight tangent field ℓ of a white key light source is visualized using LIC, and compared to the highlight curves of 3 light sources (in red, green & blue): the closer the light source direction, the better ℓ characterizes highlight shape. (d-e) Critical points through which one highlight curve goes independently of \mathbf{u} are displayed in red ($\mathbf{h}^T \mathbf{n} > 0.9995$).

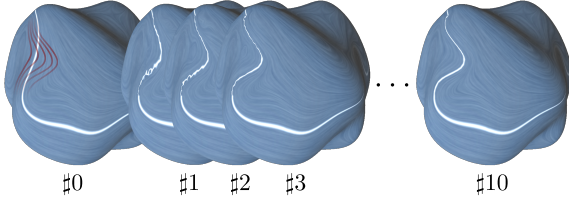


Figure 6: Starting with an initial direction field and a target highlight tangent field (shown with red curves), we apply a few iterations of our solver which here converges in 10 steps.

This equation is linear with respect to the unknown gradient $\nabla\varphi_i$, which is thus estimated through a standard linear regression on the vertices j in the 1-ring neighborhood of i :

$$\nabla\varphi_i = \frac{1}{k} \arg \min_{\phi \in \mathbb{R}^2} \sum_j \left(\phi^T \mathbf{p}_j^i - \sin(k(\varphi_j^i - \varphi_i)) \right)^2, \quad (10)$$

where $k = 1$ (resp. $k = 2$) is used for asymmetric (resp. symmetric) BRDFs (see Section 4.4), and $\mathbf{p}_j^i = [\mathbf{t}_i \ \mathbf{b}_i]^T (\mathbf{p}_j - \mathbf{p}_i)$ and φ_j^i correspond to \mathbf{p}_j and φ_j expressed in the tangent frame at vertex i . Although the former is obtained through a simple orthogonal projection, the latter has to be parallel transported to avoid any in-plane rotation. We refer to the method of Knöppel et al. [KCPS13], which has the advantage of being linear with respect to φ . Note that since parallel transport preserves lengths and we use unit vectors, we have $\|\hat{\mathbf{u}}_i(\mathbf{x})\| = 1$ for all vertices i in Equation 9.

Dozens of approaches have been developed to compute the Weingarten map \mathbf{W}_i . As before, we directly estimate the coefficients of \mathbf{W}_i by fitting the quadratic polynomial $\mathbf{x}^T \mathbf{W}_i \mathbf{x}$ to the elevation component $\mathbf{n}_i^T (\mathbf{p}_j - \mathbf{p}_i)$ of neighboring vertices. This approach can be seen as a variant of the osculating jet method [CP03].

4.3. Non-linear optimization

We now have all the ingredients to deal with the minimization of Equation 8. This is a non-linear least-square optimization problem that we solve using a Levenberg-Marquardt (LM) method. Such a solver requires the Jacobian of $\mathbf{E}(\varphi)$, which in our case is a very sparse matrix with off-diagonal non-zeros coming from the expression of $\nabla\varphi_i$. This matrix has the same structure as the Laplacian matrix, a fact

that we exploit to speed up the computation of the Jacobian at each solver iteration. In our experiment, we found the LM method to perform considerably better compared to methods that only work with the gradient, such as the conjugate-gradient or L-BFGS [BLNZ95].

As with all non-convex energies, ours require a proper initial solution. Since our general goal is to manipulate existing highlights, we simply pick the initial direction field prior to editing. Moreover, in order to avoid the introduction of singularities in the edited direction field, we apply one pass of Laplacian smoothing on φ^* at each step. This interleaved strategy permits to introduce large variations in the field if desired. We tried the common strategy of adding Laplacian regularization terms but it has the drawback of over-smoothing results as it conflicts the optimization target. In our case, continuity is conditioned by continuity of the target highlights tangents. Figure 6 depicts a few iterations of our solver, which usually converges in less than 10 iterations.

The following pseudo-code summarizes our approach:

```

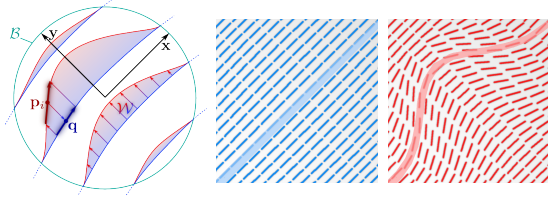
 $\ell^* \leftarrow \text{edit}(\ell)$ 
 $\varphi^* \leftarrow \text{polar}(\mathbf{u})$  ▷ eq. 16
while ( $\|\mathbf{E}(\varphi)\| < \varepsilon$ ) do ▷ eq. 7, 8
     $\varphi^* \leftarrow LM_1(\varphi^*, \mathbf{E}(\varphi), \text{Jac}(\mathbf{E}(\varphi)))$ 
     $\varphi^* \leftarrow \text{smooth}(\varphi^*)$ 
end while
return  $\mathbf{u}(\varphi^*)$  ▷ eq. 6
    
```

Here, LM_1 stands for one iteration of the Levenberg-Marquardt algorithm. Note that $\nabla\varphi^*$ (eq. 10) must be re-computed at each iteration.

4.4. Symmetric BRDF

So far, we have considered general anisotropic BRDFs whose orientation is determined by \mathbf{u} . However, many BRDF models are defined via micro-facet distributions that exhibit a central symmetry (e.g., [War92, LKK00, APS00]). In terms of BRDF orientation, this means that \mathbf{u} can safely be considered as non-oriented; or equivalently, that φ is defined modulo π .

There are two practical consequences to the use of a non-oriented \mathbf{u} : we must set $k = 2$ for the computation of $\nabla\varphi$ in Equation 10 and we must employ tensor interpolation inside triangles when rendering with such BRDFs. Other than that, our approach remains the same.



(a) Warping approach (b) Before warping (c) After warping

Figure 7: (a) We illustrate how the warping \mathcal{W} of potential highlights inside a support \mathcal{B} is applied by modifying tangents using the inverse and Jacobian of \mathcal{W} . (b-c) When applied to a mesh, the input highlight tangent field (in blue) is deformed to yield the expected target field (in red).

5. Highlight manipulation tools

A solution for specifying target highlight tangents would be to rely on tangent field design methods (e.g., [KCPS13]). However, we believe that directly providing tangents might not always be straightforward for users, and could make it difficult to find a proper initialization for optimization. We choose to adopt an alternative “shading-based” editing solution [GZ08].

Instead of manipulating highlight tangents, we thus provide tools that give the *illusion of deforming highlight curves* themselves. Each tool has a brush support $\mathcal{B} \subset \mathcal{M}$ so that only points $\mathbf{p}_i \in \mathcal{B}$ are affected. As a result, the entry and exit points of a specular highlight curve going through \mathcal{B} remain invariant; the same is true for critical points.

We demonstrate this approach with a small set of tools that could be extended to a larger toolbox. All tools use the same planar parametrization for the brush support, obtained by fitting a plane to points in \mathcal{B} . Their effects on simple highlights are rendered using a few key light sources and the Ashikhmin BRDF model with a Gaussian micro-facet distribution [APS00]. We overlay colored highlights on top of their tangent fields for visualization purposes, with the necessary drawback of altering material appearance.

Warp tools. We first propose a class of tools that deform highlights via a warping operator $\mathcal{W} : \mathcal{B} \rightarrow \mathcal{B}$. Provided that \mathcal{W} is differentiable and bijective, it is possible to find tangents that yield the expected warped result. This is shown in Figure 7, where the target highlight tangent at point \mathbf{p}_i is obtained as $\ell_i^* = \mathbf{J}_{\mathcal{W}}(\mathbf{q}) \ell_{\mathbf{q}}$, with $\mathbf{q} = \mathcal{W}^{-1}(\mathbf{p}_i)$, $\ell_{\mathbf{q}}$ the highlight tangent at \mathbf{q} and $\mathbf{J}_{\mathcal{W}}$ the Jacobian of \mathcal{W} . We illustrate this approach with two operators that use a common fall-off function: $w(x, y) = (1 - x^2 - y^2)^2$. The *bend* tool warps highlights in the positive y direction of the brush; it is defined by $\mathcal{W}_b(x, y) = [x, y + w(x, y)/2]$ and is shown in Figure 8. The *expand* tool warps highlights along y in both directions; it is defined by $\mathcal{W}_e(x, y) = [x, y + w(x, y) \text{sign}(y) \sqrt{\sin(y)}/2]$ and is shown in Figure 9. Observe how slight differences in \mathbf{u} lead to large differences in ℓ in both cases. In its use, the tool is similar to the method of Ritschel et al. [ROTS09] except that ours is tied to an inverse process whereas theirs only post-processes output radiance.

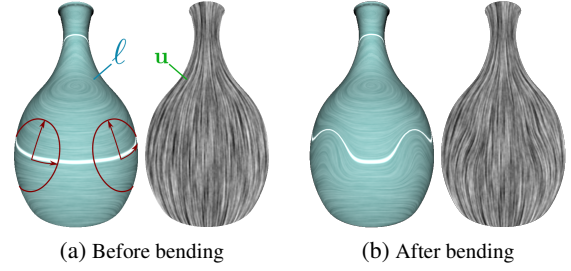


Figure 8: (a) We show how the *bend* tool acts upon an anisotropic highlight, which yields (b) a new highlight tangent field with the expected warped highlight.

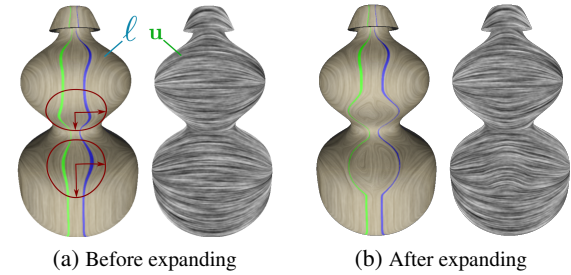


Figure 9: (a) Two highlights are edited at once with the *expand* tool, which (b) modifies their trajectories.

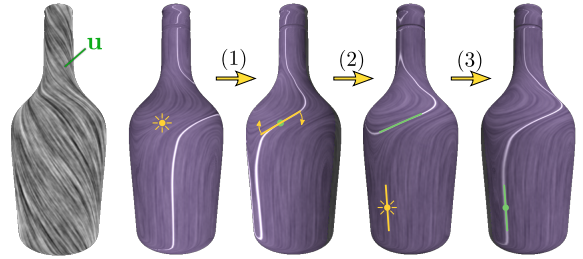


Figure 10: Starting from an existing \mathbf{u} , the *light* tool retrieves a light source: 1) by picking a point through which the highlight passes (preserving ℓ); 2) by specifying ℓ locally; 3) by choosing a novel control point for the highlight.

Light tool. Instead of manipulating \mathbf{u} by considering a fixed reference light source ω_{in} , our *light* editing tool does the reverse. It assumes a fixed direction field and looks for a light source that could have produced a given highlight tangent vector. In practice, and as demonstrated in the supplemental video, the user draws such a vector at a vertex i , which gives a value for ℓ_i that we use to recover ω_{in} in two steps. First, we know from Section 3.1 that \mathbf{h} must lie in the plane perpendicular to \mathbf{u}_i , which restricts candidates to a 1D family of half-way vectors parametrized by θ_n . We further restrict this family by $\mathbf{n}_i^T \omega_{in} > \epsilon$ to ensure that vertex i will remain sufficiently lit (we use $\epsilon = \pi/8$). Second, using a brute-force

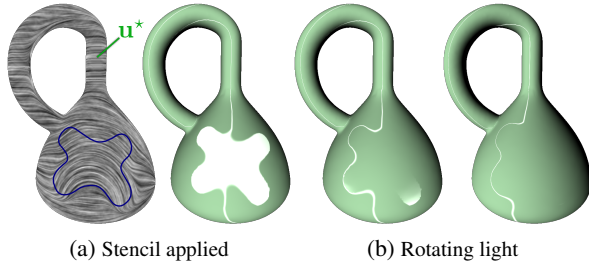


Figure 11: (a) The *stencil* tool creates a highlight region using a specific direction field \mathbf{u}^* ; (b) as a result, highlights are very sensitive to rotations of lighting.

search, we look for the θ_h that leads to the highlight tangent closest to the desired one, using Equation 7 again. The tool may be used in a variety of ways, as shown in Figure 10: to position key lights when designing a scene; to select a reference light source in an existing environment lighting; or to move an existing key light while minimizing changes in ℓ (in this case, we directly use ℓ_i as a target in the 1D search).

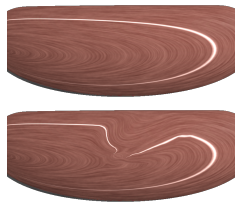
Stencil tool. We also provide a *stencil* tool that relies on the highlight regions of Section 3.1 instead of the highlight tangents of Section 3.2. The idea here is to directly set directions $\mathbf{u}_i^* = \pm \mathbf{h} \times \mathbf{n}_i$ for points inside the brush support. To avoid the creation of steep edges, we smooth \mathbf{u}_i^* using Laplacian regularization as in the optimization of Section 4.3. As shown in Figure 11, the resulting highlights may take on any desired shape but as expected quickly revert to curve-like boundaries when either the light or view changes.

6. Performance and additional results

For all our results, our solver performed at interactive rates on an Intel I7-3820 3.6GHz CPU, using a single core. Performance is linear in terms of iterations; for each iteration, timings depend on the number of edited vertices, as detailed in Figure 12a. In our experience, this is sufficient for editing purposes, even though the performance could further be improved through parallelization. A common problem of non-linear optimization is that the solver may be stuck in local minima, as shown in Figure 12b. When this issue arises, we run the optimization again with slightly different targets.

# vert.	Jacob.	Solver	Total
14	0.6	0.8	1.4
32	1.4	0.4	1.8
89	3.0	2.7	5.7
217	3.9	23.4	27.3
420	9.5	55.2	64.7
656	14.2	120.5	134.7

(a) Timings (in ms)



(b) Failure case

Figure 12: (a) Timings per iteration, with detailed measurements for the construction of the Jacobian and the solver itself. (b) A failure case showing the solver stuck in a local minimum (top: input; bottom: edit).

Even with a limited set of tools, our optimization-based approach allows users to more naturally control the shape of anisotropic highlights. This is shown in Figure 1, where our method is applied to a familiar house-hold item in different viewing and lighting configurations. In comparison, directly manipulating BRDF orientations would be purely impractical, as shown in the supplemental video. Figure 13 demonstrates our approach on a more complex object for a variety of anisotropic BRDFs. Since our optimization only considers BRDF orientations, it is valid for various types of materials, except for slanted velvet as expected. As it considers potential anisotropic highlights, the results of our method remain compelling in scenes with global illumination. This is demonstrated for three points of view in Figure 14, and on an animation sequence in the supplemental video.

7. Conclusion and future work

The main insight of our work is the extension of the analysis of Lu et al. [LKK99, LKK00] to arbitrary surfaces. From a theoretical standpoint, it leads to the characterization of *potential* highlight tangent fields, given in vectorial and angular forms in Equations 4 and 7, respectively. We use them in practice to orient anisotropic BRDFs from specified highlight tangents through optimization, and we believe our approach to be the first to address this problem. We also use our insights for selecting light sources responsible for an anisotropic highlight. We expect additional applications to be discovered in the future.

Our approach has inherent limitations though. Critical points cannot be edited since they depend on the light-view configuration, not on BRDF orientations. Highlight editing is thus significantly restricted when dealing with complex surfaces where several critical points occur. Materials with slanted micro-facet distributions like velvet [LK98] are not well accounted by our current model since they are made of non-tangential fibers. In future work, we would like to extend our differential analysis to the special case of slanted fibers. A limitation that we share with Lu et al. is that we consider lighting and viewing to be distant. This approximation is only local per surface point though, and it raised no issue in our experiments.

Our optimization technique could also be improved in a number of ways. We could consider using the amount of anisotropy of a BRDF model in addition to its orientation, which might permit to blend between different highlight patterns when varying viewing or lighting conditions. An interesting problem would then be to control these patterns with respect to the course of the sun, with applications in architecture [WJB*13]. Finally, we hope that this work will eventually permit the fabrication of objects (e.g., brushed metals, shiny fabric) with controlled anisotropic highlight appearance. This will likely require to incorporate additional energies related to fabrication constraints.

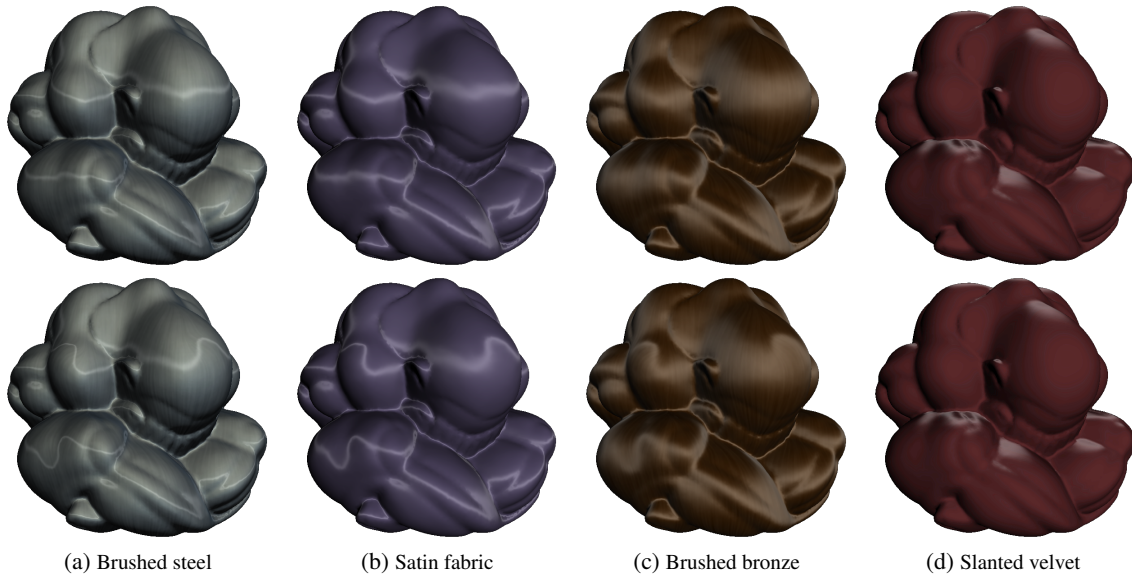


Figure 13: Top row: a statue model using a simple input direction field, lit with a single key light and rendered with four materials. Bottom row: the same model rendered with identical materials and lighting, using a single edited direction field. Highlight shape is preserved across materials produced by Ashikhmin & Shirley BRDF model [AS00], except for slanted velvet [APS00] for which highlights are concentrated around contours.



Figure 14: A scene rendered with global illumination, composed of a satin cloth [APS00], a bright bronze plate [AS00], and a steel vase [War92]. Left column: three different views using simple direction fields. Right column: same views using edited direction fields. We used the `light` tool to select key lights, `warp` tools on the vase and (quite subtly) on the cloth, and the `stencil` tool on the plate.

Acknowledgements

We would like to thank Laurent Belcour, Romain Vergne, Nicolas Mellado and Patrick Reuter for preliminary reviews and helpful discussions. We are grateful to the anonymous reviewers of this paper and members of the PRISM network for their valuable remarks. This research has been supported by the ALTA project (ANR-11-BS02-006).

References

- [Ade01] ADELSON E. H.: On seeing stuff: the perception of materials by humans and machines. *Proceedings of SPIE 4299* (2001), 1–12. 1
- [APS00] ASHIKHMIN M., PREMOŽE S., SHIRLEY P.: A microfacet-based brdf generator. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques* (2000), SIGGRAPH '00, pp. 65–74. 2, 5, 6, 8
- [AS00] ASHIKHMIN M., SHIRLEY P.: An Anisotropic Phong BRDF Model. *Journal of Graphics Tools* 5, 1 (2000), 25–32. 1, 8
- [BAERD08] BEN-ARTZI A., EGAN K., RAMAMOORTHI R., DURAND F.: A precomputed polynomial representation for interactive BRDF editing with global illumination. *ACM Transactions on Graphics* 27, 2 (Apr. 2008), 1–13. 2
- [BAOR06] BEN-ARTZI A., OVERBECK R., RAMAMOORTHI R.: Real-time BRDF editing in complex lighting. In *ACM Transactions on Graphics* (July 2006), vol. 25, ACM, p. 945. 2
- [BLNZ95] BYRD R. H., LU P., NOCEDAL J., ZHU C.: A limited memory algorithm for bound constrained optimization. *SIAM J. Sci. Comput.* 16, 5 (Sept. 1995), 1190–1208. 5
- [CDS10] CRANE K., DESBRUN M., SCHRÖDER P.: Trivial connections on discrete surfaces. *Computer Graphics Forum (SGP)* 29, 5 (2010), 1525–1533. 2
- [CL93] CABRAL B., LEEDOM L. C.: Imaging vector fields using line integral convolution. In *Proceedings of the 20th annual conference on Computer graphics and interactive techniques* (1993), SIGGRAPH '93, ACM, pp. 263–270. 4
- [CP03] CAZALS F., POUGET M.: Estimating differential quantities using polynomial fitting of osculating jets. In *Proceedings of the 2003 Eurographics/ACM SIGGRAPH symposium on Geometry processing* (2003), SGP '03, Eurographics Association, pp. 177–187. 5
- [CPK06] COLBERT M., PATTANAIK S., KR J.: The BRDF-Shop interface. *Science*, February (2006), 30–36. 2
- [dC76] DO CARMO M. P.: *Differential Geometry of Curves and Surfaces*. Prentice-Hall, Englewood Cliffs, NJ, 1976. 3, 9
- [FDA03] FLEMING R. W., DROR R. O., ADELSON E. H.: Real-world illumination and the perception of surface reflectance properties. *Journal of Vision* 3 (2003), 347–368. 1
- [FS07] FISHER M., SCHRÖDER P.: Design of tangent vector fields. *ACM Transactions on Graphics (TOG)* (2007). 2
- [GZ08] GINGOLD Y., ZORIN D.: Shading-based surface editing. *ACM Transactions on Graphics (TOG)* (2008). 2, 6
- [KCPS13] KNÖPPEL F., CRANE K., PINKALL U., SCHRÖDER P.: Globally optimal direction fields. *ACM Trans. Graph.* 32, 4 (2013). 2, 5, 6
- [LK98] LU R., KOENDERINK J. J.: Optical properties (bidirectional reflection distribution functions) of velvet. *Applied Optics* 37 (1998), 5974–5984. 7
- [LKK99] LU R., KOENDERINK J., KAPPERS A. M. L.: Specularities on surfaces with tangential hairs or grooves. In *Proceedings of the Seventh IEEE International Conference on Computer Vision* (1999), vol. 1, pp. 2–7 vol.1. 3, 7

- [LKK00] LU R., KOENDERINK J. J., KAPPERS A. M. L.: Specularities on Surfaces with Tangential Hairs or Grooves. *Computer Vision and Image Understanding* 1, 3 (2000), 320–335. 2, 3, 5, 7
- [Nic77] NICODEMUS F.: *Geometrical Considerations and Nomenclature for Reflectance*. NBS Monograph. U.S. Government Printing Office, 1977. 1
- [OMSI07] OKABE M., MATSUSHITA Y., SHEN L., IGARASHI T.: Illumination brush: Interactive design of all-frequency lighting. In *Proceedings of the 15th Pacific Conference on Computer Graphics and Applications* (2007), PG '07, IEEE Computer Society, pp. 171–180. 2
- [Pel10] PELLACINI F.: envyLight: an interface for editing natural illumination. In *ACM Transactions on Graphics (TOG)* (2010), vol. 29, ACM, p. 34. 2
- [ROTS09] RITSCHEL T., OKABE M., THORMÄHLEN T., SEIDEL H.-P.: Interactive reflection editing. *ACM Trans. Graph.* 28, 5 (Dec. 2009), 129:1–129:7. 2, 6
- [TS67] TORRANCE K. E., SPARROW E.: Theory for off-specular reflection from roughened surfaces. *JOSA* (1967). 2
- [War92] WARD G.: Measuring and modeling anisotropic reflection. *ACM SIGGRAPH Computer Graphics* 26, 2 (1992), 265–272. 5, 8
- [WJB*13] WANG J., JIANG C., BOMPAS P., WALLNER J., POTTMANN H.: Discrete line congruences for shading and lighting. *Computer Graphics Forum* 32/5 (2013). 7
- [ZMT06] ZHANG E., MISCHAIKOW K., TURK G.: Vector field design on surfaces. *ACM Trans. Graph.* 25, 4 (2006), 1294–1326. 2

Appendix

We provide details on how Equation 4 can be re-written in terms of angles φ to yield Equation 7.

We write the tangent frame $\mathbf{U} = [\mathbf{u} \ \mathbf{v} \ \mathbf{n}]$ in terms of $\bar{\mathbf{u}}(\varphi)$ and $\bar{\mathbf{v}}(\varphi)$ and we omit the dependence on φ for clarity:

$$\mathbf{U} = [\mathbf{t} \ \mathbf{b} \ \mathbf{n}] \begin{bmatrix} \bar{\mathbf{u}} & \bar{\mathbf{v}} & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Incorporating this formula inside Equation 4 while dropping the explicit zeros, we obtain:

$$\ell(\varphi) = \pm [\mathbf{t} \ \mathbf{b}] [\bar{\mathbf{u}} \ \bar{\mathbf{v}}] \begin{bmatrix} (-\nabla \cdot \bar{\mathbf{u}}) \bar{\mathbf{v}}^T & -\tau(\bar{\mathbf{u}}) \\ (\nabla \times \bar{\mathbf{u}}) \bar{\mathbf{v}}^T & \kappa(\bar{\mathbf{u}}) \end{bmatrix} [\mathbf{t} \ \mathbf{b} \ \mathbf{n}]^T \mathbf{h}. \quad (11)$$

Observing from Equation 6 that the Jacobian of $\bar{\mathbf{u}}$ verifies $\mathbf{J} = \bar{\mathbf{v}} \nabla \varphi^T$, the divergence of $\bar{\mathbf{u}}$ in terms of φ is given by $\nabla \cdot \bar{\mathbf{u}} = \text{tr}(\mathbf{J}) = \bar{\mathbf{v}}^T \nabla \varphi$. Since the curl of $\bar{\mathbf{u}}$ also corresponds to the divergence of $\bar{\mathbf{v}}$, with a similar reasoning we obtain:

$$\begin{bmatrix} -\nabla \cdot \bar{\mathbf{u}} \\ \nabla \times \bar{\mathbf{u}} \end{bmatrix} = \begin{bmatrix} -\bar{\mathbf{v}}^T \\ \bar{\mathbf{u}}^T \end{bmatrix} \nabla \varphi. \quad (12)$$

Moreover, the surface curvature κ and torsion τ in the $\bar{\mathbf{u}}(\varphi)$ direction are commonly computed using the 2×2 Weingarten map \mathbf{W} [dC76], which yields:

$$\begin{bmatrix} -\tau(\bar{\mathbf{u}}) \\ \kappa(\bar{\mathbf{u}}) \end{bmatrix} = \begin{bmatrix} -\bar{\mathbf{v}}^T \\ \bar{\mathbf{u}}^T \end{bmatrix} \mathbf{W} \bar{\mathbf{u}}. \quad (13)$$

Finally, plugging Equations 12 and 13 in Equation 11 while observing that $[\bar{\mathbf{u}} \ \bar{\mathbf{v}}] [-\bar{\mathbf{v}} \ \bar{\mathbf{u}}]^T$ is a rotation of $\pi/2$, we obtain:

$$\ell(\varphi) = \pm [\mathbf{b} \ -\mathbf{t}] \begin{bmatrix} \nabla \varphi \bar{\mathbf{v}}^T & \mathbf{W} \bar{\mathbf{u}} \end{bmatrix} [\mathbf{t} \ \mathbf{b} \ \mathbf{n}]^T \mathbf{h}.$$