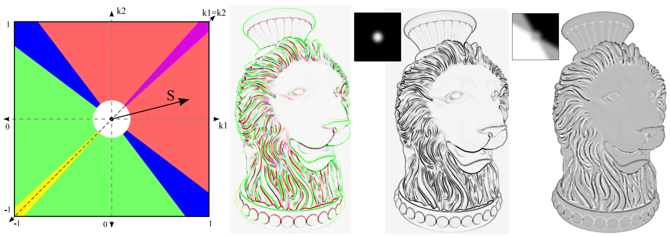
Shading with Apparent Relief

Romain Vergne, Pascal Barla, Xavier Granier, Christophe Schlick

Shading with Apparent Relief

Romain Vergne* Pascal Barla* Xavier Granier* Christophe Schlick*

INRIA Bordeaux University



(a) (b) (c) (d)

Figure 1: *(a)* ***Shape descriptor domain****. we show the color code used to visualize our shape descriptor: planes are in white, caps (in yellow) and cups (in pink) are located around the symmetry axis, and saddles (in blue) separate convex (in green) from concave regions (in red). The axis of the parameter space correspond to principal curvatures k_1 and k_2. (b) Apparent Relief. (c) planar regions are filtered out to reveal the entire features of the object. (d) concavities are drawn in white, convexities in black and planar- and saddle-like regions in gray. The respective ARMs are shown in the top left corner of the figures (0.4 Mtri / 244 fps).*

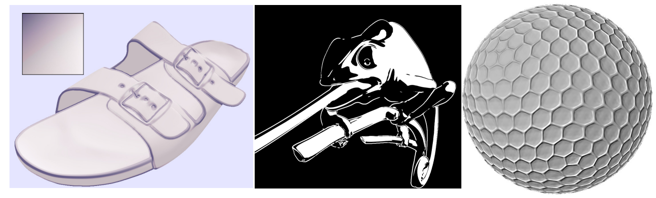
1 Introduction

Shape depiction is an important dimension of image creation. For example, techniques are used to remove ambiguities in scientific illustrations, or to create more legible representations in paintings and drawings. Previous approaches focus on a set body of techniques: line-based rendering. Such techniques generate some *shape cues* to depict the object characteristics that correspond to *discontinuities* of shape features. Many artists rather depict an object's shape through shading. They have to use other kinds of shape cues that correspond to *continuous* variations of shape features to convey more subtle information about shape and integrate it seamlessly into conventional lighting. Instead of detecting sharp *discontinuities* as in line-based rendering, we thus seek a set of *continuous* cues that have to be defined for each pixel of an image. To this end, we introduce an intermediate representation that we call Apparent Relief to assist the user. It is a view-dependent shape descriptor, from which continuous shape cues are easily extracted, and which gives rise to stylized shading-based shape depictions. By construction, it is free of temporal coherence artifacts and naturally leads to *automatic* Levels-of-Detail (LOD) effects. Our approach is simple to implement, runs in real-time on modern graphics hardware, and allows a user selection of features. We illustrate its potential using several shading styles.

2 Apparent Relief descriptor

Our solution to shading-based shape depiction is to manipulate continuous cues through a shape descriptor inspired from [Koenderink and van Doorn 1992]: for each pixel, we extract a vector $\mathbf{S}$, which corresponds to a position in the shape descriptor (Figure 1(a)). The direction $\mathbf{D}$ gives information about surface convexity and is computed for each vertex in object-space using the vector $\mathbf{K} = (k_1, k_2)$ composed of principal curvatures: $\mathbf{D} = \mathbf{K}/||\mathbf{K}||$. When $||\mathbf{K}|| = 0$ (i.e., the center of the shape descriptor domain, corresponding to planar regions on the surface), we set $\mathbf{D} = (0,0)$. The length $\mathbf{L}$ of the vector $\mathbf{S}$, which represents the curvedness information, is directly computed in image space to provide view-dependency and automatic LODs. It is computed as the amount of variation of normals in an extended pixel neighborhood using Gaussian derivatives of normal images transformed to camera space. Our Apparent Relief Descriptor consists in convexity and curvedness information: $\mathbf{S} = \mathbf{D} * \mathbf{L}$. Figure 1(b) shows an example where $\mathbf{S}$ is computed for each pixel, and displayed using the color code given in (a).

*e-mail: {vergne|barla|granier|schlick}@labri.fr



(a) 0.3 Mtri / 250 fps (b) 0.15 Mtri / 250 fps (c) 1 Mtri / 40 fps

Figure 2: *Cartoon shading, counter shading, and exaggerated shading.*

$\mathbf{S}$ can easily be used as a texture coordinate to lookup a relief value $r \in [0, 1]$ in a texture called Apparent Relief Map (ARM) which can be modified in real-time by the user. By directly assigning this value to the pixel, we obtain a simple **cel-shading** style (Figures 1(c) and (d)). The user can easily draw into the texture to select and enhance the desired shape cues. As shown in Figure 2(a), conventional **cartoon-shading** may easily be modified using X-Toon [Barla et al. 2006] to create more complex styles. In our case, the detail coordinate corresponds to the relief value. **Counter-shading** is a drastic shading style which uses only black and white colors. We use relief values obtained from the ARM as contrast values (Figure 2(b)). Counter shading is thus defined by a linear interpolation between a diffuse and inverted diffuse intensity: $I = (1-r)\mathbf{n}\cdot\mathbf{l} + r(1-\mathbf{n}\cdot\mathbf{l})$, where n and l define the current point's normal and light unit vectors. I is then thresholded at 0.5. Figure 2(c) shows how the ARM is combined with **exaggerated shading** [Rusinkiewicz et al. 2006] to control which cues may be enhanced (here convex regions).

3 Discussion and future work

Our new shape descriptor called Apparent Relief, based on a combination of object-space and image-space measurements, provides a simple and flexible approach to select continuous and view-dependent shape cues with automatic LOD functionalities. A painting interface allows the user to directly draw into the ARM to choose which cues are selected and which ones are filtered with a direct feed-back. The user can also use the inverse approach and enhance some cues by simply clicking on a pixel of the rendered image. The ARM is then automatically updated with the corresponding vector $\mathbf{S}$ and a chosen color. We would like to improve the interface to provide a faster and a more flexible control for the user. We also plan to deal with dynamic scenes on which principal curvatures need to be recomputed at each frame.

References

Barla, P., Thollot, J., and Markosian, L. 2006. X-toon: An extended toon shader. In *NPAR*, ACM.

Koenderink, J. J., and van Doorn, A. J. 1992. Surface shape and curvature scales. *Image Vision Comput. 10*, 8, 557–565.

Rusinkiewicz, S., Burns, M., and DeCarlo, D. 2006. Exaggerated shading for depicting shape and detail. *ACM Trans. Graph (Proc. SIGGRAPH) 25*, 3 (July).