Improved Geometric Specular Antialiasing (Supplemental Document)

Yusuke Tokuyoshi
SQUARE ENIX CO., LTD.
tokuyosh@square-enix.com

Anton S. Kaplanyan
Facebook Reality Labs
kaplanyan@fb.com

1 Non-Axis-Aligned Anisotropic BRDF

Shadowing-masking Function. The Smith masking function [Smi67] is defined as $G_1(\mathbf{i},\mathbf{h}) = \frac{\chi^+(\mathbf{i}\cdot\mathbf{h})}{1+\Lambda(\mathbf{i})}$, where $\chi^+(\mathbf{i}\cdot\mathbf{h})$ is the Heaviside function: 1 if $\mathbf{i}\cdot\mathbf{h} > 0$ otherwise 0. $\Lambda(\mathbf{i})$ is a function which depends on the NDF model. The height-correlated masking-shadowing function [Hei14] is given as

$$G_2(\mathbf{i},\mathbf{o}) = \frac{\chi^+(\mathbf{i}\cdot\mathbf{h})\,\chi^+(\mathbf{o}\cdot\mathbf{h})}{1+\Lambda(\mathbf{i})+\Lambda(\mathbf{o})}.$$

In this paper, $\Lambda(\mathbf{o})$ for the anisotropic GGX NDF model is described in the later paragraphs.

Axis-aligned Anisotropic GGX BRDF The axis-aligned anisotropic GGX NDF is defined as follows:

$$D(\mathbf{h}) = \frac{\chi^+(h_z)}{\pi\alpha_x\alpha_y\left(\frac{h_x^2}{\alpha_x^2}+\frac{h_y^2}{\alpha_y^2}+h_z^2\right)^2}.$$

For this NDF, the masking-shadowing function is obtained using the following function:

$$\Lambda(\mathbf{o}) = -0.5 + \frac{\sqrt{\alpha_x^2 o_x^2 + \alpha_y^2 o_y^2 + o_z^2}}{2|o_z|},$$

where $[o_x, o_y, o_z]$ is the outgoing direction $\mathbf{o}$ in tangent space.

Non-axis-aligned Anisotropic GGX BRDF For shading antialiasing, we use the 2×2 roughness matrix $\mathbf{A}$ instead of α_x and α_y. The anisotropic NDF can be generalized using this matrix [Hei14] as follows:

$$D(\mathbf{h}) = \frac{\chi^+(h_z)}{\pi\sqrt{\det(\mathbf{A})}\left([h_x,h_y]\mathbf{A}^{-1}[h_x,h_y]^{\mathrm{T}}+h_z^2\right)^2}.$$

For this NDF, the masking-shadowing function is obtained using the following function:

$$\Lambda(\mathbf{o}) = -0.5 + \frac{\sqrt{[o_x, o_y]\mathbf{A}[o_x, o_y]^{\mathrm{T}} + o_z^2}}{2|o_z|}.$$

For this microsurface model, the slope of a microsurface is stretched in the directions of the eigenvectors of the roughness matrix $\mathbf{A}$. The stretching scale for each eigenvector is the reciprocal square root of the eigenvalue of $\mathbf{A}$.

Practical Implementation. The determinant $\det(\mathbf{A})$ can produce a large precision error due to floating point arithmetic, especially when using an elongated kernel for NDF filtering. To improve the numerical stability, this paper clamps $\det(\mathbf{A})$ by a small value τ for NDF:

$$D(\mathbf{h}) = \frac{\chi^+(h_z)}{\pi\sqrt{\max(\det(\mathbf{A}), \tau)}\left([h_x, h_y]\mathbf{A}^{-1}[h_x, h_y]^{\mathrm{T}} + h_z^2\right)^2}.$$

To compute $\mathbf{A}^{-1}$, we also use this clamped determinant as follows:

$$\mathbf{A}^{-1} = \frac{\mathrm{adj}(\mathbf{A})}{\max(\det(\mathbf{A}), \tau)}.$$

For NDF filtering, since $\sqrt{\det(\mathbf{A})}$ must be equal or greater than the original squared roughness parameter, we use $\tau = \alpha_x^2\alpha_y^2$.

2 Derivation of the Jacobian Matrix

Let ψ_x be an angle on the great circle passing through the halfvector $\mathbf{h}$ and normal $\mathbf{n}$, and ψ_y be an angle on the great circle passing through the halfvector $\mathbf{h}$ and $\frac{\mathbf{n}\times\mathbf{h}}{\|\mathbf{n}\times\mathbf{h}\|}$: then its Cartesian coordinate is given as

$$\begin{aligned} m_x &= \cos\psi_y \sin\psi_x, \\ m_y &= \sin\psi_y, \\ m_z &= \cos\psi_y \cos\psi_x. \end{aligned} \tag{1}$$

Thus, the Jacobian matrix of the transformation from $[\psi_x, \psi_y]$ to $[m_x, m_y]$ at $\psi_x = 0$ and $\psi_y = 0$ is yielded as

$$\begin{aligned} J_{\circ\to\perp^m} &= \begin{bmatrix} \frac{\partial m_x}{\partial \psi_x} & \frac{\partial m_x}{\partial \psi_y} \\ \frac{\partial m_y}{\partial \psi_x} & \frac{\partial m_y}{\partial \psi_y} \end{bmatrix} \\ &= \begin{bmatrix} \cos\psi_y\cos\psi_x & -\sin\psi_y\sin\psi_x \\ 0 & \cos\psi_y \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}. \end{aligned} \tag{2}$$

The tangent-space halfvector can be represented using a polar coordinate system $[\theta, \phi]$. Using this θ and this ϕ, the rotation from the local-space halfvector to tangent-space halfvector is given by

$$\begin{bmatrix} h_x \\ h_y \\ h_z \end{bmatrix} = \begin{bmatrix} \cos\theta\cos\phi & -\sin\phi & \sin\theta\cos\phi \\ \cos\theta\sin\phi & \cos\phi & \sin\theta\sin\phi \\ -\sin\theta & 0 & \cos\theta \end{bmatrix} \begin{bmatrix} m_x \\ m_y \\ m_z \end{bmatrix}, \tag{3}$$

where $[m_x, m_y, m_z] = [0, 0, 1]$ (i.e., $\psi_x = 0$ and $\psi_y = 0$). Therefore, the Jacobian matrix of the orthographic projection is derived as

$$\begin{aligned} J_{\circ\to\perp} = J_{\perp^m\to\perp} J_{\circ\to\perp^m} &= \begin{bmatrix} \frac{\partial h_x}{\partial m_x} & \frac{\partial h_x}{\partial m_y} \\ \frac{\partial h_y}{\partial m_y} & \frac{\partial h_y}{\partial m_y} \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} \cos\theta\cos\phi & -\sin\phi \\ \cos\theta\sin\phi & \cos\phi \end{bmatrix} \\ &= \frac{1}{\sqrt{1-h_z^2}} \begin{bmatrix} h_x h_z & -h_y \\ h_y h_z & h_x \end{bmatrix}. \end{aligned} \tag{4}$$

The slope of the halfvector is given as

$$\begin{bmatrix} h_x^{\parallel} & h_y^{\parallel} \end{bmatrix} = \begin{bmatrix} -\frac{h_x}{\sqrt{1-h_x^2-h_y^2}} & -\frac{h_y}{\sqrt{1-h_x^2-h_y^2}} \end{bmatrix}. \tag{5}$$

Therefore, the Jacobian matrix of the transformation from the projected unit disk to slope space is yielded as follows:

$$J_{\perp\to\parallel} = \begin{bmatrix} \frac{\partial h_x^{\parallel}}{\partial h_x} & \frac{\partial h_x^{\parallel}}{\partial h_y} \\ \frac{\partial h_y^{\parallel}}{\partial h_x} & \frac{\partial h_y^{\parallel}}{\partial h_y} \end{bmatrix} = -\frac{1}{h_z^3} \begin{bmatrix} 1-h_y^2 & h_x h_y \\ h_x h_y & 1-h_x^2 \end{bmatrix}. \tag{6}$$

Hence, the Jacobian matrix of the transformation from spherical space to slope space is obtained as

$$J_{\circ\to\parallel} = J_{\perp\to\parallel} J_{\circ\to\perp} = -\frac{1}{h_z^2\sqrt{1-h_z^2}} \begin{bmatrix} h_x & -h_y h_z \\ h_y & h_x h_z \end{bmatrix}. \tag{7}$$

References

[Hei14] Eric Heitz. Understanding the masking-shadowing function in microfacet-based BRDFs. *Journal of Computer Graphics Techniques (JCGT)*, 3(2):48–107, 2014.

[Smi67] B. G. Smith. Geometrical shadowing of a random rough surface. *IEEE Trans. Antennas and Propagation*, 15(5):668–671, 1967.