Curvature Depended Local Illumination Approximation of Ambient Occlusion

Tomohito Hattori　Hiroyuki Kubo　Shigeo Morishima
Waseda University

1. Introduction[1]

This paper discusses an approach for computing the ambient occlusion by curvature depended approximation of occlusion. Ambient occlusion is widely used to improve the realism of fast lighting simulation. The ambient occlusion is defined as follows.

$$L_{abocc}(x) = \frac{1}{\pi}\int_{\Omega_+} V(x,\omega)\cos\theta \quad d\omega \qquad (1)$$

V(x, ω) is the visibility function in direction *ω* from vertex location *x*. *θ* is the angle between vertex normal and *ω*. Ω_+ is a hemisphere domain of integration defined by the normal. While this method enables effective lighting simulation by considering surrounding occlusion, previous works [Miguel et al. 2008 : Janne et al. 2005], needs great computation cost, because the integration is solved by monte-carlo collision detection. By contrast, we approximate occlusion by curvature to aim to reduce the cost.

2. Shape Geometry Approximation

Precisely, the previous ambient occlusion needs monte-carlo collision detection because it is difficult to predict neighborhood surface from target pixel. This is a serious problem, and it is not solved in SSAO yet. Then assuming that a function that approximate geometry shape makes easy to achieve occlusion, we will acquire a function z = f(*x,y*) .To acquire concrete expression of f(x,y), we calculate Taylor series expansion of f(x,y). Considering from quadratic terms of the Taylor series, we acquire eq.(2) using maximum principal curvature as κ_1 and minimum principal curvature as κ_2, because, the quadratic terms are equal to curvature.

$$f(x,y) = \frac{1}{2}\left(\kappa_1 x^2 + \kappa_2 y^2\right) \qquad (2)$$

To calculate neighborhood occlusion, eq.(2) is effective function, because, this function is extremely equal to neighborhood shape.

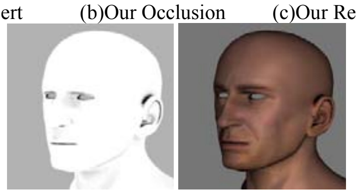
3. Approximation of Ambient Occlusion

In this section we introduce a method to calculate occlusion from eq.(2). In previous techniques, they have to calculate the ratio of occluded ray in Ω_+ to acquire occlusion. By contrast, we can acquire the ratio of occluded area on Ω_+ analytically, because we already acquired neighborhood shape accuracy according to eq.(2). Thereby, we calculate occluded area by the eq.(2) surface to acquire occlusion. The equation is as follows.

$$L_{abocc}(\kappa_1,\kappa_2) = \int_0^{2\pi}\int_0^{\theta} r^2\varphi \sin\theta' d\theta' d\varphi \qquad (3)$$

$$\theta = \arccos\left(\frac{-1 \pm \sqrt{1 + A^2}}{A}\right), \qquad A = R\left(\kappa_1 \cos^2\varphi + \kappa_2 \sin^2\varphi\right)$$

(a)Lambert (b)Our Occlusion (c)Our Rendering



(d)Previous Technique (e)Previous Rendering

Figure 1 Occlusion by Our Technique

4. Implementation

In this section we present an efficient implementation of our curvature depended ambient occlusion. We compute occlusion from eq.2 in advance and make a look up table (LUT), because it needs much computation cost to calculate eq.(3) to acquire exact solution. We write this LUT into a texture that have principal curvatures as UV-vector, and we apply this texture when we render ambient occlusion. Then our result is in Figure 1. We measured the run-time performance on a desktop PC with Intel Core 2 Duo 2.66GHz processor NVIDIA GeForce GTX 285 GPU. The rendering speed is more than 1000fps. According to our result, we can render ambient occlusion faster and easier than past technique with equal quality.

5. Results and Discussions

Past ambient occlusion technique needs much computation because of global illumination model. For this reason, we proposed a new, simple ambient occlusion technique that is approximated into local illumination. Our technique can render ambient occlusion by 1000fps in real-time.

In this work, we pre-compute curvature with CPU before rendering. So, real-time curvature calculation from vertex by GPU is our future work to treat a non-rigid object.

References

HAYDEN, L. 2002. Production-Ready Global Illumination. *Siggraph2002*.

MIGUEL, S. Real-Time Depth Buffer Based Ambient Occlusion. Game Developer Conference, 2008

JANNE K., SAMULI L. 2005. Ambient Occlusion Field. *Siggraph 2005*

---

[1] †httr-zodiac@fuji.waseda.jp
‡shigeo@waseda.jp

SIGGRAPH 2010, Los Angeles, California, July 25 – 29, 2010.
ISBN 978-1-4503-0210-4/10/0007