

# Rendering Details on Anisotropic Surfaces with Approximate MDF Rotation

Young-Min Kang  
Dept. of Game Engineering  
Tongmyong University  
Busan, Korea  
Email: ymkang@tu.ac.kr

Chang-Sik Cho  
ETRI  
Daejeon, KOREA  
Email: cscho@etri.re.kr

**Abstract**—In this paper, we propose an approach to rendering mesoscopic details on anisotropic reflectance surfaces. In order to represent the geometric details in realtime rendering applications, normal mapping or normal perturbation methods are widely used. In our observation, however, the normal perturbation on anisotropic reflectance surface usually introduces unrealistic reflection artifacts. In order to avoid such artifacts and improve the rendering quality of normal mapped anisotropic surface, we propose an MDF (microfacet distribution function) rotation method that efficiently express the reflectance change caused by perturbing the normal vectors. Moreover, the MDF rotation effect can be easily obtained without actually rotating the MDF. We also proposed an efficient approximation of the MDF rotation, and the method does not have to compute the perturbed tangent space. The proposed method can be easily implemented with GPU programs, and works well in realtime environments.

## I. INTRODUCTION

Early idea about microfacet-based rendering was introduced by Torrance and Sparrow [16]. In this methods, the surface to be rendered was assumed as a collection of very small facets and each facet has its own orientation and reflects like a mirror. The reflectance property of this surface model is determined by the microfacet distribution function (MDF).

The microfacet-based rendering model has been continuously improved to represent various materials. Techniques for controlling the roughness of the surface were also introduced [6], [5], and those methods were also improved by Cook and Torrence [7].

In daily observation, we can easily notice that metallic surfaces show anisotropic reflectance because of the subtle normal perturbation on the surface. There have been various techniques for representing the anisotropic reflectance [11], [18], [14]. Ashikhmin and Shirley proposed an anisotropic reflection model with intuitive control parameters [3], [4]. Their model is successfully utilized to express the surface with brushed scratches.

Wang *et al.* proposed a method that approximates the measured BRDF (bidirectional reflectance distribution function) with multiple spherical lobes [17]. Although this method is capable of reproducing various materials including metallic surface, it has a serious disadvantage in that expensively measured BRDF is required.

Although there have been many approaches to the representation of metallic surface [15], relatively little attention has

been given to the effect of the normal perturbation on the surface. In order to represent the geometric details in realtime rendering applications, normal mapping or normal perturbation methods are widely used. In our observation, however, the normal perturbation on anisotropic reflectance surface usually introduces unrealistic reflection artifacts. In order to avoid such artifacts and improve the rendering quality of normal mapped anisotropic surface, we propose a method that rotates the MDF (microfacet distribution function) of the surface in accordance with the perturbation of the normal vectors. Our contribution is demonstrated in Fig. 1. In the figure, (a) and (b) show the effect of simple traditional normal mapping on Ward and Ashikhmin anisotropic surface. Fig. 1 (c) and (d) show the effect of our improved normal mapping techniques on the same surfaces. The improved normal mapping can express the geometric details more apparently.

The proposed method can be also applied to woven fabric rendering. Yasuda *et al.* proposed a fabric shading model by applying anisotropic reflectance [19]. However, the method is not capable of rendering the close-up scene. Adabala *et al.* proposed a fabric rendering method that can be applied to both distant and close-up observations of woven surface [2], [1]. In this method, however, realistic light reflection was not the major concern. Sattler *et al.* employed BTF (bidirectional texture function) proposed by Dana *et al.* [8] to render photorealistic woven fabric [13]. However, one needs to prepare texture data for all kinds of fabric which will be possibly used in rendering. The method proposed in this paper can be also successfully applied to realtime rendering of woven fabric.

## II. VISUAL ARTIFACTS OF NORMAL MAPPED ANISOTROPIC SURFACE

The reflectance property of microfacet-based surface model is determined by the microfacet distribution function (MDF)  $D(\mathbf{h})$  which gives the probability that a microfacet is oriented to the direction  $\mathbf{h}$ . Ashikhmin *et al.* proposed an anisotropic reflectance model, and their MDF  $D_A(\mathbf{h})$  can be expressed as follows:

$$D_A(\mathbf{h}) = \frac{\sqrt{(e_x + 1)(e_y + 1)}}{2\pi} (\mathbf{h} \cdot \mathbf{n})^{e_x \cos^2 \phi + e_y \sin^2 \phi} \quad (1)$$

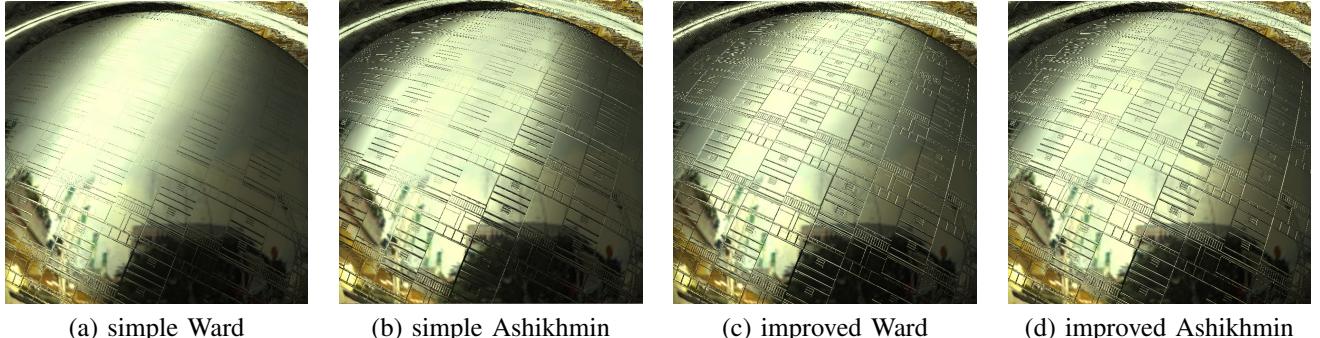


Fig. 1. The improved normal mapping on anisotropic reflectance surface: (a) & (b) simple normal mapping on Ward surface and Ashikhmin surface respectively, (c) & (d) improved normal mapping with our method on Ward and Ashikhmin surfaces respectively.

where  $\mathbf{n}$  is the normal vector at the point to be rendered. The actual parameter  $\mathbf{h}$  in the MDF is the half way vector between the incident light direction and outgoing viewing direction.  $e_x$  and  $e_y$  are parameters that control the anisotropy of the reflection, and  $\phi$  is the azimuthal angle of the half way vector.

There have been continuous efforts to represent higher geometric complexity with simple mesh by perturbing the normal vectors [12], [9], [10]. Bump mapping is well known in graphics literature, normal mapping is an improved method which does not compute normal vectors during the rendering phase [12].

Heidrich and Seidel applied Blinn-Phong shading to the normal mapped geometry [9]. Their method is successful only when the reflection is isotropic. However, the normal mapping on anisotropic reflection surface, unfortunately, cannot reproduce the original anisotropic reflectance on the distorted surface because the normal mapping or other normal vector perturbation methods only change the normal vector  $\mathbf{n}$ . Let us denote the perturbed normal vector as  $\tilde{\mathbf{n}}$ . The normal-perturbed MDF of Ashikhmin-Shirley model  $\tilde{D}_A(\mathbf{h}, \tilde{\mathbf{n}})$  can then be rewritten as follows:

$$\tilde{D}_A(\mathbf{h}, \tilde{\mathbf{n}}) = \frac{\sqrt{(e_x + 1)(e_y + 1)}}{2\pi} (\mathbf{h} \cdot \tilde{\mathbf{n}})^\epsilon \quad (2)$$

where  $\epsilon$  denotes  $e_x \cos^2 \phi + e_y \sin^2 \phi$ .

Fig. 2 shows the MDF computed with Eq. 2 and perturbed normal vectors. The cross mark in the figure indicates the perturbed normal. The top row of Fig. 2 shows isotropic MDF when the normal vector is perturbed. As shown in the figure, Eq. 2 produces reasonable deformed MDF for the isotropic MDF. However, the simple normal perturbation is not successful with anisotropic MDFs. The bottom row of Fig. 2 shows the results when we employed an anisotropic MDF. The results show that simple normal perturbation approach is hopelessly unsuccessful to preserve the original reflection property.

The situation is even worse for the Ward BRDF. The original MDF  $D_W(\mathbf{h}, \mathbf{n})$  and normal-perturbed MDF  $\tilde{D}_W(\mathbf{h}, \tilde{\mathbf{n}})$  of Ward model can be expressed as follows:

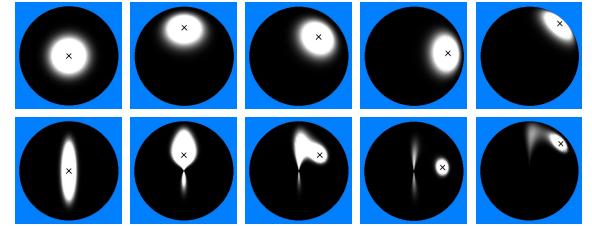


Fig. 2. Ashikhmin MDF with perturbed normal vectors: (top row) perturbation with isotropic MDF and (bottom row) perturbation with anisotropic Ashikhmin MDF.

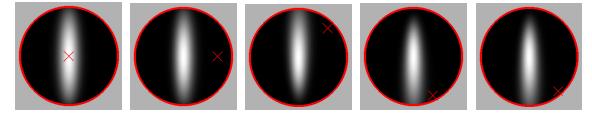


Fig. 3. Ward MDF with perturbed normal vectors

$$D_W(\mathbf{h}, \mathbf{n}) = e^{-2 \frac{\mathbf{h}_x/\alpha_x + \mathbf{h}_y/\alpha_y}{1+\mathbf{h}\cdot\mathbf{n}}} \quad (3)$$

$$\tilde{D}_W(\mathbf{h}, \tilde{\mathbf{n}}) = e^{-2 \frac{\mathbf{h}_x/\alpha_x + \mathbf{h}_y/\alpha_y}{1+\mathbf{h}\cdot\tilde{\mathbf{n}}}}.$$

The visualized MDF of Ward surface with perturbed normal vector is shown in Fig. 3. As shown in the figure, perturbed normal vectors could not effectively deform the reflection property of the Ward MDF. The cross mark in the MDF function is the location of the perturbed normal. In the Ward model, the anisotropic MDF dose not change much even though the normal vector is severely perturbed.

### III. IMPROVED NORMAL MAPPING WITH MDF ROTATION

In order to overcome the limitation of the simple normal mapping on anisotropic reflection surface, the MDF should be properly deformed with the original anisotropic property maintained. Fig. 4 shows the MDF rotation concept. The original MDF is defined in tangent space with three axes,  $\mathbf{u}, \mathbf{v}, \mathbf{w}$ . In this original frame,  $\mathbf{u}$  is  $(1, 0, 0)^T$ ,  $\mathbf{v}$  is  $(0, 1, 0)^T$ , and  $\mathbf{w}$  is  $(0, 0, 1)^T$ . The normal vector is coincident with  $\mathbf{w}$ . When we apply normal mapping, the normal should be perturbed, and will not be  $(0, 0, 1)^T$  any more. Suppose that the perturbed normal  $\tilde{\mathbf{n}}$  is  $(\Delta x, \Delta y, \sqrt{1 - \Delta x^2 - \Delta y^2})$ . Let us denote the deformed MDF as  $D'(\mathbf{h})$ . We can easily derive

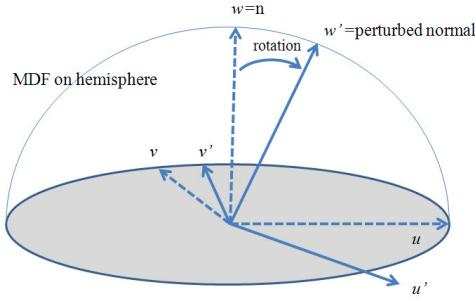


Fig. 4. MDF rotation concept.

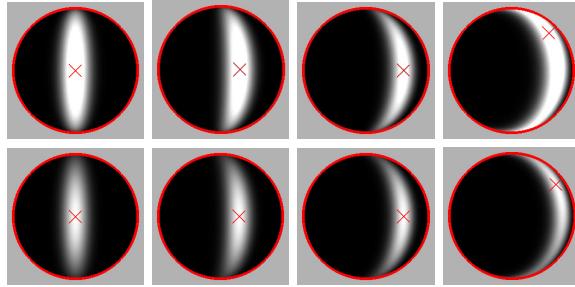


Fig. 5. MDF rotation of (top row) Ashikhmin-Shirley BRDF ( $e_x = 1.0, e_y = 40.0$ ), and (bottom row) Ward BRDF ( $\alpha_x = 0.1, \alpha_y = 0.5$ ).

$D'(\mathbf{h})$  with the MDF rotation concept shown in Fig. 4. We simply rotate the original MDF defined on the hemisphere to the new coordinate system with axes  $\mathbf{u}'$ ,  $\mathbf{v}'$ ,  $\mathbf{w}'$ .

Since it is obvious that  $\mathbf{w}'$  is the perturbed normal, we can easily determine the  $\mathbf{w}'$ , obtain orthogonal axes  $\mathbf{u}'$  and  $\mathbf{v}'$ , and the rotation matrix as follows:

$$\begin{aligned} \mathbf{w}' &= \tilde{\mathbf{n}} = (\Delta x, \Delta y, \sqrt{1 - \Delta x^2 - \Delta y^2})^T \\ \mathbf{u}^{tmp} &= (0, 1, 0)^T \times \mathbf{w}' \\ \mathbf{v}' &= \frac{\mathbf{w}' \times \mathbf{u}^{tmp}}{|\mathbf{w}' \times \mathbf{u}^{tmp}|} \\ \mathbf{u}' &= \frac{\mathbf{v}' \times \mathbf{w}'}{|\mathbf{v}' \times \mathbf{w}'|} \\ \mathbf{R} &= [\mathbf{u}', \mathbf{v}', \mathbf{w}']. \end{aligned} \quad (4)$$

The rotation matrix  $\mathbf{R}$  transforms the original coordinate frame in accordance with the perturbed normal as shown in Fig. 4. Let us denote the transformation of a vector  $\mathbf{p}$  according to the perturbed normal as follows:

$$\mathcal{T}(\mathbf{p}, \tilde{\mathbf{n}}) = \mathbf{R}\mathbf{p}. \quad (5)$$

Fig. 5 shows the result when the original MDF is rotated with the transformation  $\mathbf{R}$ , and we denote this MDF as  $D'(\mathbf{h})$ . However, it is obvious that computing the rotated MDF at each sampling point on the surface is extremely inefficient. Explicit deformation of the MDF is only a conceptual process. In the actual rendering process, we never compute  $D'(\mathbf{h})$ . Only the

original MDF  $D(\mathbf{h})$  is used with the inverse transformation  $\mathcal{T}^{-1}(\mathbf{p}', \tilde{\mathbf{n}})$ . In other words, we conceptually employ  $D'(\mathbf{h})$  for the normal mapped surface, but actually use  $D(\mathcal{T}^{-1}(\mathbf{h}, \tilde{\mathbf{n}}))$  which has the equivalent value.

The inverse transformation of Eq. 5 can be easily obtained as follows:

$$\mathcal{T}^{-1}(\mathbf{p}', \tilde{\mathbf{n}}) = \mathbf{R}^T \mathbf{p}. \quad (6)$$

Now we can simply calculate  $D(\mathcal{T}^{-1}(\mathbf{h}, \tilde{\mathbf{n}}))$  to compute the MDF at the point where the normal vector is perturbed as  $\tilde{\mathbf{n}}$ .

It should be noted that the MDF with the inverse transformation, i.e.,  $D(\mathcal{T}^{-1}(\mathbf{h}, \tilde{\mathbf{n}}))$ , still remain in the original MDF space. The normal vector is always  $(0, 0, 1)$  in tangent space. Therefore, the dot product of any vector  $\mathbf{v}$  and the normal vector  $\mathbf{n}$  (i.e.,  $\mathbf{v} \cdot \mathbf{n}$ ) is simply the  $z$  component of the vector,  $\mathbf{v}_z$ , and the actual MDF for Ashikhmin-Shirley model we used is as follows:

$$\begin{aligned} D'_A(\mathbf{h}, \tilde{\mathbf{n}}) &= \\ D_A(\mathcal{T}^{-1}(\mathbf{h}, \tilde{\mathbf{n}}), \mathbf{n}) &= \frac{\sqrt{(e_x+1)(e_y+1)}}{2\pi} \mathcal{T}^{-1}(\mathbf{h}, \tilde{\mathbf{n}})_z^\epsilon. \end{aligned} \quad (7)$$

We can easily implement the improved normal mapping for Ward surface by defining the MDF of Ward model  $D'_W(\mathbf{h}, \tilde{\mathbf{n}})$  as follows:

$$\begin{aligned} D'_W(\mathbf{h}, \tilde{\mathbf{n}}) &= \\ D_W(\mathcal{T}^{-1}(\mathbf{h}, \tilde{\mathbf{n}}), \mathbf{n}) &= e^{-2 \frac{\frac{\mathbf{h}_x}{\alpha_x} + \frac{\mathbf{h}_y}{\alpha_y}}{1 + \mathcal{T}^{-1}(\mathbf{h}, \tilde{\mathbf{n}})_z}}. \end{aligned} \quad (8)$$

Fig. 6 shows the effect of the MDF rotation by comparing the specular reflections on the illusory bumps. The bumpy illusion on the surface shown in Fig. 6 (a) and (b) are generated only with normal mapping method on Ward and Ashikhmin surfaces respectively. Fig. 6 (c) and (d) are generated with MDF rotation techniques on the same surfaces. As shown in the figure, simple normal mapping with original MDF cannot produce anisotropic reflectance on the bumpy illusion. Even worse, the shapes of the specular reflection areas are weirdly distorted on some bumps. The deformed MDF removes such disadvantages, and the anisotropic reflectance is well preserved on each illusory bump, and no weird shapes are found.

In our observation, the accurate  $\mathbf{R}^T$  is not required for plausible rendering of mesoscopic details on the surface. In order to alleviate the computational burden, we used an approximate rotation matrix  $\tilde{\mathbf{R}}^T$  as follows:

$$\begin{aligned} \mathbf{w}' &= \tilde{\mathbf{n}} = (\tilde{\mathbf{n}}_x, \tilde{\mathbf{n}}_y, \tilde{\mathbf{n}}_z)^T \\ \mathbf{u}' &= (0, 1, 0)^T \times \mathbf{w}' = (\tilde{\mathbf{n}}_z, 0, -\tilde{\mathbf{n}}_x)^T \\ \mathbf{v}' &= \mathbf{w}' \times \mathbf{u}' = (0, -\tilde{\mathbf{n}}_z, \tilde{\mathbf{n}}_y)^T \\ \tilde{\mathbf{R}}^T &= \begin{bmatrix} \tilde{\mathbf{n}}_z & 0 & -\tilde{\mathbf{n}}_x \\ -\tilde{\mathbf{n}}_x \tilde{\mathbf{n}}_y & 1 - \tilde{\mathbf{n}}_y^2 & \tilde{\mathbf{n}}_y \tilde{\mathbf{n}}_z \\ \tilde{\mathbf{n}}_x & \tilde{\mathbf{n}}_y & \tilde{\mathbf{n}}_z \end{bmatrix}. \end{aligned} \quad (9)$$

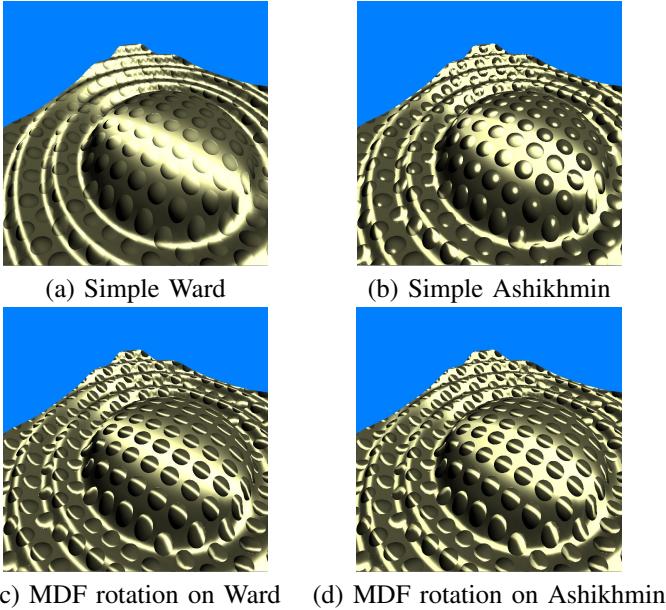


Fig. 6. Effect of MDF deformation on anisotropic reflection surfaces: (a) simple normal mapping on Ward surface, (b) simple normal mapping on Ashikhmin surface, (c) MDF rotation on Ward surface, (d) MDF rotation on Ashikhmin surface.

Let us denote  $\tilde{\mathbf{h}}$  as  $\mathcal{T}^{-1}(\mathbf{h}, \tilde{\mathbf{n}})$ . The rotated MDF  $D'_W(\mathbf{h}, \tilde{\mathbf{n}})$  is now simply computed with original MDF as follows:

$$D'(\mathbf{h}, \tilde{\mathbf{n}}) = D(\tilde{\mathbf{h}}, \mathbf{n}) = D(\tilde{\mathbf{R}}^T \mathbf{h}, \mathbf{n}). \quad (10)$$

We can easily compute  $\tilde{\mathbf{h}}$  without actual matrix operations as follows:

$$\tilde{\mathbf{h}} = \begin{bmatrix} \tilde{\mathbf{n}}_z \mathbf{h}_x - \tilde{\mathbf{n}}_x \mathbf{h}_z \\ -\tilde{\mathbf{n}}_x \tilde{\mathbf{n}}_y \mathbf{h}_x + (1 - \tilde{\mathbf{n}}_y^2) \mathbf{h}_y + \tilde{\mathbf{n}}_y \tilde{\mathbf{n}}_z \mathbf{h}_z \\ \tilde{\mathbf{n}}_x \tilde{\mathbf{n}}_x + \tilde{\mathbf{n}}_y \tilde{\mathbf{n}}_y + \tilde{\mathbf{n}}_z \tilde{\mathbf{n}}_z \end{bmatrix}. \quad (11)$$

Now we can obtain normal mapping or normal perturbation effect by simply applying the perturbed half vector shown in Eq. 11, and the computational cost can be significantly reduced.

#### IV. EXPERIMENTS

The approximate MDF rotation is more efficient than the accurate rotation. We compared the time required for MDF rotation, and the result is shown in Fig. 7. For rendering one frame shown in Fig. 8, the Accurate MDF rotation required about 54  $\mu$ sec while the approximate version required only about 11  $\mu$ sec.

The rendering results of the accurate MDF rotation and the approximate approach are compared in Fig. 8. As shown in the figure, the approximate perturbation matrix produced almost the same result as that of the accurate version.

In order to demonstrate the advantages of the proposed method, we applied the proposed method to two specific rendering applications where the MDF rotation techniques can be successfully utilized: rendering of metallic surfaces with complex geometry, and rendering of woven fabric objects. Our method is based on GPU-friendly algorithm. Therefore, we

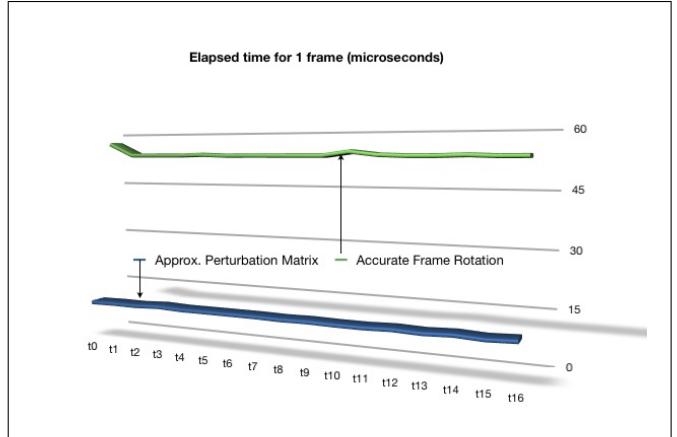


Fig. 7. Comparison of the time required for rendering one frame with accurate MDF rotation and the approximate approach

could successfully implement those rendering applications to work in realtime.

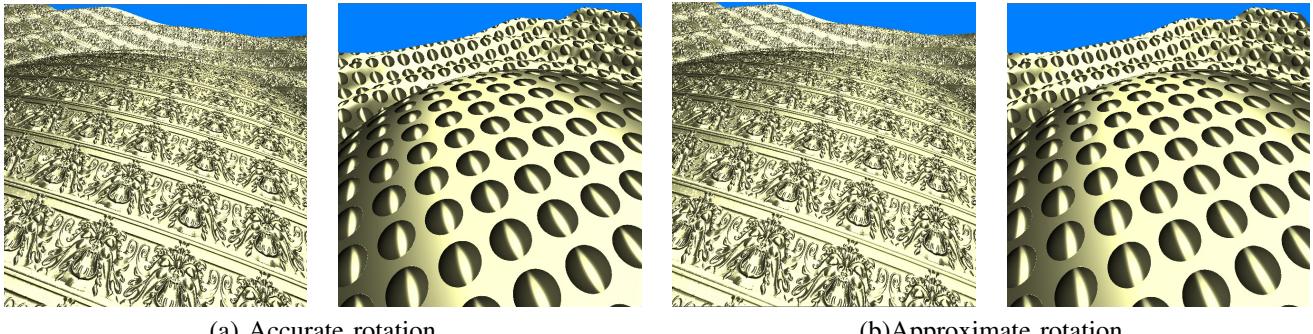
#### A. Realtime Rendering of Metallic Surface

The proposed method can be directly applied to realistic and realtime rendering of metallic surface. In order to increase the geometric complexity, we used normal map textures. The proposed method could effectively express the geometric details expected by the normal map textures. Fig. 9 compares the light scattering on normal mapped anisotropic reflection surface. Fig. 9 (a) and (b) show the rendering results where normal mapping is applied without rotating the MDF on Ward and Ashikhmin surfaces respectively. The bumpy illusion on the surface can be more clearly expressed by our method. (c) and (d) show results rendered with additional MDF rotation. As shown in the figure, the proposed method is much more expressive than the simple normal mapping in representing the geometric details expressed by normal map texture.

#### B. Realtime Rendering of Woven Fabric Objects

In order to demonstrate the rendering quality of the proposed method, we applied the method to woven fabric rendering. Woven fabric has weft and warp yarns. Because the yarns are oriented in different directions, the reflectance anisotropy is alternating according to the yarn direction. Therefore, we employed alternating anisotropy for woven fabric, and the anisotropy is determined by the underlying weave patterns. We can easily alternate the anisotropy by swapping the parameters  $e_x$  and  $e_y$ .

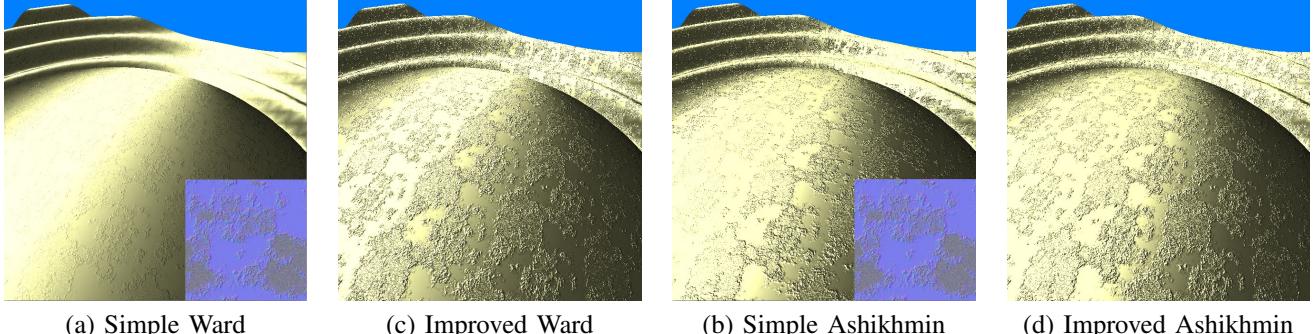
The alternating anisotropy can be efficiently and effectively utilized for describing the woven fabric reflectance. However, the alternating anisotropy cannot represent the bumpy surface of woven fabric. In order to produce realistic bumpy surface caused by woven structure, we have to perturb the normal vectors on the fabric according to the weave patterns. This is exactly the same problem as the normal mapping on anisotropic reflectance surface. In this case, however, we do not need to use any actual normal map texture because we can



(a) Accurate rotation

(b) Approximate rotation

Fig. 8. The rendering results of (a) accurate MDF rotation and (b) approximate rotation



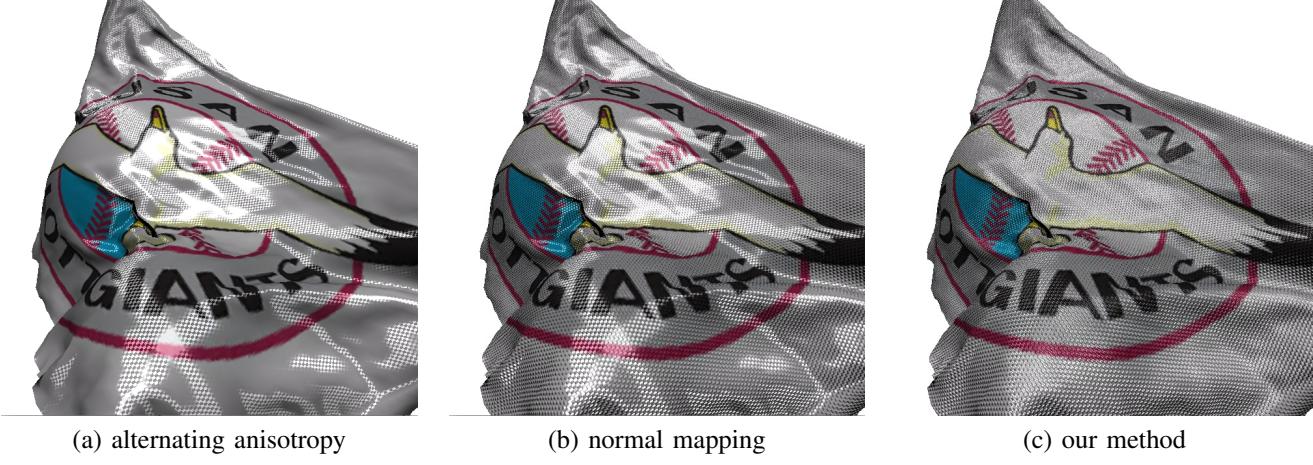
(a) Simple Ward

(c) Improved Ward

(b) Simple Ashikhmin

(d) Improved Ashikhmin

Fig. 9. Realistic realtime rendering of metallic surface: (a,b) simple mapping, (c,d) improved normal mapping with our method on Ward and Ashikhmin surfaces respectively.



(a) alternating anisotropy

(b) normal mapping

(c) our method

Fig. 10. Fabric rendering results

easily perturb the normal vectors procedurally by taking the weave patterns into account.

Fig. 10 shows the fabric rendering results with our method. In this figure, (a) shows the rendering result when the anisotropy of the reflection is simply alternated in accordance with the yarn type (weft/warp). The alternating anisotropy could express the different reflection properties of weft and warp yarns. However, in some area where the weft and warp yarns show the same reflection, the weave patterns are not visible. (b) shows the result when procedural normal perturbation for expressing the curved surface of yarns are applied. As shown in the figure, the woven yarn structure are more

clearly visible. However, the overall reflection property of the fabric surface is still too metallic. (c) is the result when our MDF rotation techniques are applied to the normal-perturbed surface. As shown in the figure, the specular reflection on the surface is smoothly and naturally diffused so that the rendering result is more realistic.

### C. Performance

The techniques proposed in this paper was implemented with OpenGL shading language, and the computing environments were Mac OS X operating system with 3.2 GHz Intel core i3 CPU, 4 Gb DDR3 RAM and ATI Radeon

Tech	Gouraud	Aniso	N-Map	MDF
Cost	1	1.03	1.55	1.56
FPS	1004	968	647	642

TABLE I

RENDERING PERFORMANCE OF THE PROPOSED METHOD COMPARED WITH OTHER REALTIME METHODS.

HD 5670 GPU. Table. I is the performance analysis of the proposed method compared with traditional approaches. The label 'Aniso' means Ashikhmin-Shirley anisotropic reflection model, 'N-map' represents normal mapping, and 'MDF' indicates the proposed MDF rotation techniques. The computational cost of Gouraud shading is taken as a unit cost, and other rendering techniques were compared with the unit cost. As shown in the figure, the proposed method is just slightly more expensive than usual normal mapping which works very well in realtime environments. The anisotropic reflectance rendering with GPU program is 1.03 times expensive, and simple normal mapping is 1.55 times expensive. As shown in the table, our method is not that expensive compared with simple normal mapping. It is only 1.56 times expensive compared with Gouraud shading. The row name "FPS" shows the number of the rendered frames per second with each method on the system mentioned before.

## V. CONCLUSION

In this paper, we proposed an improved normal mapping techniques for anisotropic reflectance surface modeled with microfacet-based BRDF. The proposed method is not dependent on the BRDF models so that it can be applied to any surface of which reflectance is described with MDF.

The proposed method can be the solution to the not-well-known problem of simple normal mapping on anisotropic reflectance surfaces. When the anisotropy of the surface is modeled with MDF and the perturbed normal vector is simply given to the function, the resulting reflectance does not preserve the original anisotropy any more. We demonstrated the weird MDF deformation by the perturbed normal vectors, and proposed more reasonable MDF rotation approach. The MDF rotation method effectively deform the MDF in accordance with the perturbed normal vectors still maintaining the anisotropic reflectance property.

We also proposed an efficient approximation of the MDF rotation in order to obtain the same effect with alleviated computational cost. Moreover, the approximate method does not have to compute the perturbed tangent space. The half-way vector in the new coordinate frame can be easily computed by considering the perturbed normal only.

The experimental results show that the proposed method can be successfully utilized to represent metallic surfaces of which geometric complexity is expressed by normal map texture. The MDF rotation approach is not only applicable to metallic surface but also applicable to various MDF-based reflectance models. In our experiments, the MDF deformation can be successfully utilized for realistic fabric rendering.

We implemented the proposed method with GPU programs, and the performance test is satisfactory enough for us to utilize the proposed method in realtime applications.

## ACKNOWLEDGMENT

This work was supported in part by the SW computing R&D program of MKE/KEIT [10035184], "Game Service Technology Based on Realtime Streaming", and also supported in part by the MKE(Ministry of Knowledge Economy), Korea, under the ITRC(Information Technology Research Center) support program supervised by the NIPA(National IT Industry Promotion Agency) (NIPA-2010-(C- 1090-1021-0006)).

## REFERENCES

- [1] N. Adabala, N. Magnenat-Thalmann, and G. Fei. Real-time rendering of woven clothes. In *Proceedings of the ACM Symposium on Virtual Reality Software and Technology*, pages 41–47, 2003.
- [2] N. Adabala, N. Magnenat-Thalmann, and G. Fei. Visualization of woven cloth. In *Proceedings of the 14th Eurographics Workshop on Rendering (ACM International Conference Proceeding Series)*, 44:178–185, 2003.
- [3] M. Ashikhmin, S. Premoze, and P. Shirley. A microfacet-based brdf generator. In *Proceedings of ACM SIGGRAPH 2000*, pages 65–74, 2000.
- [4] M. Ashikhmin and P. Shirley. An anisotropic phong brdf model. *Journal of Graphics Tools*, 5(2):25–32, 2002.
- [5] J. Blinn. Models of light reflection for computer synthesized pictures. *Proceedings of the 4th annual conference on Computer graphics and interactive techniques*, pages 192–198, 1977.
- [6] J. Blinn and M. Newell. Texture and reflection in computer generated images. *Communication of ACM*, 19(10):542–547, 1976.
- [7] R. L. Cook and K. E. Torrance. A reflectance model for computer graphics. *Computer Graphics (ACM Siggraph '81 Conference Proceedings)*, 15(3):307–316, 1981.
- [8] K. J. Dana, S. K. Nayar, B. Van Ginneken, and J. J. Koenderink. Reflectance and texture of real-world surfaces. *ACM Transactions on Graphics*, 18(1):1–34, 1999.
- [9] W. Heidrich and H.-P. Seidel. Realistic, hardware-accelerated shading and lighting. In *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques*, pages 171–178, 1999.
- [10] M. Pharr and G. Humphreys. In *Physically-based Rendering*. Elsevier (Morgan Kaufman Publishers), 2004.
- [11] M. Poulin and A. Fournier. A model for anisotropic reflection. *Computer Graphics (ACM Siggraph '90 Conference Proceedings)*, 23(4):273–282, 1990.
- [12] H. Rushmeier, G. Taubin, and A. Gueziec. Applying shapes from lighting variation to bump map capture. In *Proceedings of Eurographics Rendering Workshop '97*, pages 35–44, 1997.
- [13] M. Sattler, R. Sarlette, and R. Klein. Efficient and realistic visualization of cloth. *EGRW '03: Proceedings of the 14th Eurographics workshop on Rendering*, pages 167–177, 2003.
- [14] C. Schilick. A customizable reflectance model for everyday rendering. In *Proceedings of the 4th Eurographics Workshop on Rendering*, pages 73–84, 1993.
- [15] L. Szirmay-Kalos, T. Umenhoffer, Gustavo Patow, L. Szecsi, and M. Sbert. Specular effects on the gpu: State of the art. *Computer Graphics Forum*, 28(6):1586–1617, 2009.
- [16] K. E. Torrance and E. M. Sparrow. Theory for off-specular reflection from roughened surfaces. *Journal of Optical Society of America*, 57(9), 1967.
- [17] J. Wang, P. Ren, M. Gong, J. Snyder, and B. Guo. All-frequency rendering of dynamic, spatially-varying reflectance. In *Proceedings of ACM Siggraph Asia 2009*, pages 1–10, 2009.
- [18] G. Ward. Measuring and modeling anisotropic reflection. *Computer Graphics (ACM Siggraph '92 Conference Proceedings)*, 26(2):265–272, 1992.
- [19] T. Yasuda, Yokoi S., Toriwaki J., and Inagaki K. A shading model for cloth objects. *IEEE Computer Graphics and Applications*, 12(6):15–24, 1992.