

Paper:

Neural Network Implementation of Image Rendering via Self-Calibration

Yi Ding^{*1}, Yuji Iwahori^{*2}, Tsuyoshi Nakamura^{*1}, Lifeng He^{*3},
Robert J. Woodham^{*4}, and Hidenori Itoh^{*1}

^{*1}Nagoya Institute of Technology
Gokiso-cho, Showa-ku, Nagoya 466-8555, Japan
E-mail: ding@juno.ics.nitech.ac.jp

^{*2}Dept. of Computer Science, Chubu University
1200 Matsumoto-cho, Kasugai 487-8501, Japan

^{*3}Faculty of Information Science and Technology, Aichi Prefectural University
1522-3 Kumabari-Ibaragabasama, Nagakute, Aichi-gun, Aichi 480-1198, Japan

^{*4}University of British Columbia
Vancouver, B.C. V6T 1Z4, Canada

[Received August 26, 2009; accepted January 7, 2010]

This paper proposes a new approach for self-calibration and color image rendering using Radial Basis Function (RBF) neural network. Most empirical approaches make use of a calibration object. Here, we require no calibration object to both shape recovery and color image rendering. The neural network learning data are obtained through the rotations of a target object. The approach can generate realistic virtual images without any calibration object which has the same reflectance properties as the target object. The proposed approach uses a neural network to obtain both surface orientation and albedo, and applies another neural network to generate virtual images for any viewpoint and any direction of light source. Experiments with real data are demonstrated.

Keywords: neural network based rendering, photometric stereo, self-calibration, albedo, shape recovery

1. Introduction

Model based rendering purposes generating realistic images from the 3-D modeling of the real object. In general, 3-D modeling deals with both photometric and geometric properties such as shape, viewpoint, lighting, and albedo. Rendering is originally based on the technology of 3-D computer graphics and it has been used in graphics architecture, video games and recently in the area of computer vision and mixed reality.

In the previous approaches for shape recovery, Woodham [1] proposed photometric stereo to recover the surface orientation from shading images under three light sources. Further, an empirical photometric stereo [2] was proposed in 1994. Empirical photometric stereo uses LUT (Look Up Table) to look up the surface gradient from three image intensities from a calibration sphere with the

same reflectance properties as the target object.

Iwahori et al. [3] developed Neural Network (NN) implementation of photometric stereo. Radial Basis Function Neural Network (RBF-NN) [4] learns the mapping of triple of image intensities to the corresponding surface gradient using a calibration sphere. This means NN learns the surface reflectance property with a calibration sphere and generalizes to the target object. NN based photometric stereo [5] estimates the monochrome albedo (reflectance factor) except the surface gradients.

Further, the approach [6] recovers the color reflectance factor and surface gradients for the purpose of color image rendering, here 3-D shape and color reflectance factor are obtained by NN, then NN based image rendering is applied to generate virtual images for any viewpoint and any direction of light source.

The approach [7] proposed a method that renders virtual objects from real illumination environment. 3-D shape is obtained by the range finder. Reflectance parameters are estimated based on the Torrance-Sparrow reflectance function. [7] uses the range finder and it is difficult to realize the entire approach with only photometric environment.

Shape-from-silhouette [8] recovers the shape of an object from many images taken at many viewpoints. Shape-from-silhouette requires the observation from 360° rotation with the small rotation angle to recover the detailed shape. The method does not need to know reflectance parameters. However it has the problem to catch the local concave shape.

[9] combines the techniques of photometric stereo and motion stereo to recover both 3-D shape and surface reflectance parameters. Instead, it is required that specific reflectance function is assumed to obtain the surface reflectance parameters.

In this paper, we propose a new approach to improve neural network based rendering without any calibration sphere. Instead, the rotation of the target object itself



generates the learning data for neural network via self-calibration. Using the dichromatic reflection model, the image intensities of specular reflection and diffuse reflection are separated from the observed data. Four images of the target object under different four light sources are used to recover the shape and to generate virtual images, including recovering color reflectance factor.

2. Background

2.1. Principle of Photometric Stereo

In [5], NN based photometric stereo is used to recover both surface orientation and surface albedo. Four light source NN based photometric stereo uses a calibration sphere which has the same reflectance properties as the target object, and the following simultaneous equation holds.

$$\begin{cases} E_1(x,y) = R_1(\mathbf{n}, \rho) \\ E_2(x,y) = R_2(\mathbf{n}, \rho) \\ E_3(x,y) = R_3(\mathbf{n}, \rho) \\ E_4(x,y) = R_4(\mathbf{n}, \rho) \end{cases} \dots \dots \dots (1)$$

where (R_1, R_2, R_3, R_4) is the reflectance map, \mathbf{n} represents the surface normal vector and ρ represents the reflectance factor (albedo). In the previous approaches [5], NN learns the mapping of (E_1, E_2, E_3, E_4) to (\mathbf{n}, ρ) for a calibration sphere and generalizes the NN to the target object to recover (\mathbf{n}, ρ) . Suppose that the height function $z = F(x, y)$, then the surface gradient parameters (p, q) are represented as $(p, q) = (\frac{\partial z}{\partial x}, \frac{\partial z}{\partial y})$. In general, \mathbf{n} can be represented with x, y, z components (n_x, n_y, n_z) or (p, q) as

$$\mathbf{n} = (n_x, n_y, n_z) = \frac{(-p, -q, 1)}{\sqrt{p^2 + q^2 + 1}} \dots \dots \dots (2)$$

2.2. Dichromatic Reflection Model

The dichromatic reflection model describes that image intensity E consists of two components, one of which is the diffuse reflection component R_d and the other of which is the specular reflection component R_m . In the diffuse reflection, the light is reflected in all directions. The color of the diffuse reflection represents the color of object, while the color of specular reflection represents the color of light source. The colors of reflection light is given by the two kinds of colors but it is assumed that the white color light source is used here.

The parameters R_d (diffuse component) and R_m (specular component) represent the mixing ratios of the dichromatic reflection model. The mixing ratio depends on the surface normal vector \mathbf{n} , the light source direction vector \mathbf{s} and the viewing direction vector \mathbf{v} at each point on the object. Here E_w represents the intensity of light source and it is assumed to be 1.

$$E = R_d E_w \rho + R_m E_w = R_d(\mathbf{n}, \mathbf{s})\rho + R_m(\mathbf{n}, \mathbf{s}, \mathbf{v}) \quad (3)$$

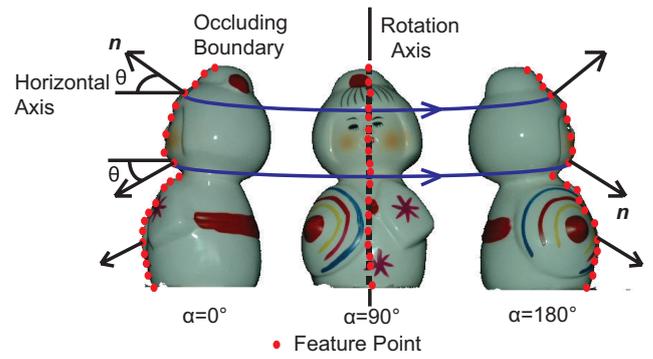


Fig. 1. Self-calibration through rotation.

3. Self-Calibration and Neural Network Learning

3.1. Self-Calibration with Rotation

Empirical photometric stereo is an approach to recover the shape of the target object using a calibration object with known shape. Here, it is assumed that a calibration object has the same reflectance property as the target object. Since a sphere has all possible surface gradients, a sphere is usually used as a calibration object. However, it is difficult to prepare the calibration sphere which has the same reflectance properties as the target object in general. To achieve the same condition as the calibration sphere, the self-calibration is introduced to obtain the reflectance properties by rotating the target object itself (as shown in Fig. 1).

The target object is rotated from 0° to 359° . The object images are obtained under each of four light sources. At the points on the occluding boundary of target object, the surface normal is perpendicular to both the tangent to the occluding boundary and the viewing direction. The surface normals of those points on the occluding boundary are uniquely determined and can be geometrically calculated. Those points are tracked during rotation and the corresponding surface normals at the rotation angle α can also be calculated. Among those points, some feature points are selected with the uniform probability according to the calculated surface normals. A set of surface normals and corresponding observed image intensities of selected feature points is used for neural network learning.

Gaussian sphere is defined as a virtual sphere with its radius $R = 1$. The surface normals is represented as the various points on the Gaussian sphere. Each point on the Gaussian sphere is projected onto the tangent plane defined as (f, g) space with the stereographic projection. The values of (f, g) take the region within a circle of radius 2 for all points on the Gaussian sphere [10].

During rotation, the feature points on the occluding boundaries are selected and tracked with every 1° . Suppose a feature point on the occluding boundaries, the point is tracked geometrically during rotation. Tracked point on the Gaussian sphere is mapped on the point inside the circle with the radius 2 in (f, g) plane as shown in Fig. 2.

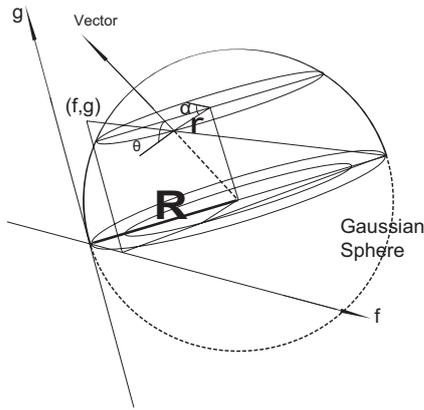


Fig. 2. Gaussian sphere and projection of feature point onto (f, g) space.

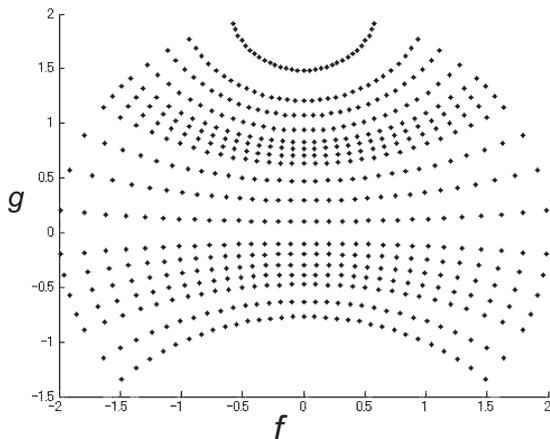


Fig. 3. Plot example of feature points in (f, g) space.

The radius of the circle, r , represents the horizontal distance at each point on the occluding boundary from the rotation axis, where $r = R \cos \theta = R \cos \theta$. θ represents the angle between the normal vector of the feature point and the horizontal axis.

From the relation of similarity for the triangles in **Fig. 2**, (f, g) of the current feature point is determined using the current rotation angle α , R and r as

$$f = \frac{2Rr \cos \alpha}{R + r \sin \alpha}$$

$$g = \pm \frac{\sqrt{(f^2 + 4R^2)(R^2 - r^2)}}{\sqrt{(R + r \sin \alpha)^2 + (r \cos \alpha)^2}} \quad \dots \quad (4)$$

g takes plus or minus value. If the vector \mathbf{n} is over the horizontal axis, g takes plus value. If the vector \mathbf{n} is under the horizontal axis, g takes minus value (as shown in **Fig. 1**). Plot example of the feature points in (f, g) space using Eq. (4) is shown in **Fig. 3**.

The values of (p, q) become infinite on the occluding boundary. Except the points on the occluding boundary, the corresponding (p, q) are given by

$$p = \frac{4f}{4 - f^2 - g^2}, \quad q = \frac{4g}{4 - f^2 - g^2} \quad \dots \quad (5)$$

for points where $(4 - f^2 - g^2) \neq 0$.

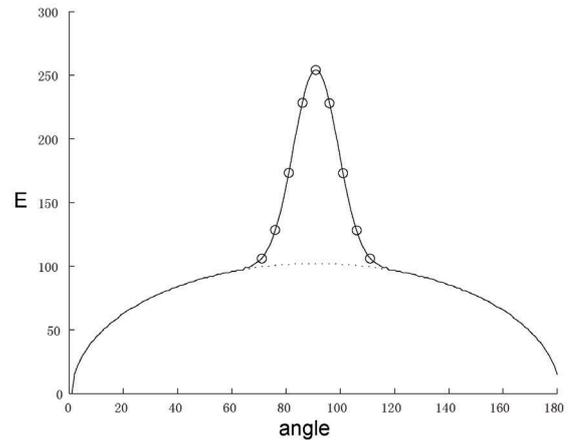


Fig. 4. Separation of diffuse and specular components.

3.2. NN for Estimating Color Reflectance Factor

When the neural network based rendering is used to generate virtual images for any viewpoint and any direction of light source, it is necessary to obtain the color reflectance factors (color albedo) of the object. To recover the color reflectance factor of all points of the object, NN learning data are prepared from the feature points and the learned NN is used to the remaining other points in the generalization. For any feature point, the color (RGB components) reflectance factor is calculated from Eq. (3) as

$$(E_R - E_{Rm})/E_{Rd} = \rho_R \quad \dots \quad (6)$$

$$(E_G - E_{Gm})/E_{Gd} = \rho_G \quad \dots \quad (7)$$

$$(E_B - E_{Bm})/E_{Bd} = \rho_B \quad \dots \quad (8)$$

When the specular components (E_{Rm}, E_{Gm}, E_{Bm}) can be separated from (E_R, E_G, E_B) based on the dichromatic model, the color reflectance factor (color albedo) (ρ_R, ρ_G, ρ_B) can be estimated since the diffuse components (E_{Rd}, E_{Gd}, E_{Bd}) are based on the cosine of the incident angle i , where i is the angle between the surface normal vector \mathbf{n} and the light source direction vector \mathbf{s} of any feature point.

The color albedo (ρ_R, ρ_G, ρ_B) can be calculated using Eqs. (6)-(8) for only the feature points. So, we use the NN to estimate color reflectance factors including other remained points of the object.

When the target object is rotated from 0° to 180° , the image intensities E of feature point are shown with the real line in **Fig. 4**.

When E suddenly changes for some threshold value, the brighter points are separated from E as E_m . The points with specular components are plotted with 'o' marker in **Fig. 4** as an example. The remaining data are recognized as the components $E_d \rho$ and the spline interpolation is applied to these data. As a result, the components $E_d \rho$ are separated from E . These components are shown with the broken line in **Fig. 4**. Here, since the diffuse components

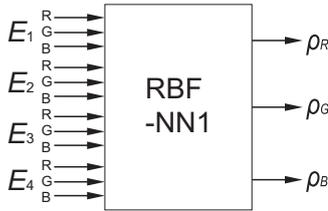


Fig. 5. NN for estimating color reflectance factor.

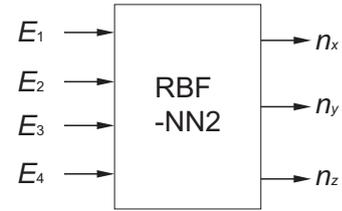


Fig. 6. NN for surface normal estimation.

E_d are calculated by $\cos i$, the color reflectance factor ρ of the feature point are calculated by $\rho = (E - E_m)/E_d$. This method is applied to each of RGB components using Eqs. (6)-(8).

The color reflectance factor of the feature points is calculated using Eqs. (6)-(8) and these data are prepared as the learning data of NN. The color reflectance factor of the other points on the target object is still unknown and they cannot be prepared as the learning data. To recover the color of all points on the target object, the learning data of the feature points is synthesized by multiplying random number using Eq. (9).

$$\begin{aligned} \rho' &= \{\rho'_R, \rho'_G, \rho'_B\} \\ &= \{\rho_R \times rand_1, \rho_G \times rand_2, \rho_B \times rand_3\}. \end{aligned} \quad (9)$$

Here, the symbol ' (dash) represents the learning data with adding the random number.

Uniform random color reflectance factors $(\rho'_R, \rho'_G, \rho'_B)$ are generated using Eq. (9), where $(rand_1, rand_2, rand_3)$ is the randomized real value between 0 and 1. The corresponding image intensities (E'_R, E'_G, E'_B) for the selected feature points are given with the randomized real value $(rand_1, rand_2, rand_3)$ according to the corresponding values of $(\rho'_R, \rho'_G, \rho'_B)$ under 4 light source directions. The mapping of $(E'_{1R}, E'_{1G}, E'_{1B}, E'_{2R}, E'_{2G}, E'_{2B}, E'_{3R}, E'_{3G}, E'_{3B}, E'_{4R}, E'_{4G}, E'_{4B})$ to $(\rho'_R, \rho'_G, \rho'_B)$ is input to the NN learning.

To estimate color reflectance factor, the learning data $(E'_{1R}, E'_{1G}, E'_{1B}, E'_{2R}, E'_{2G}, E'_{2B}, E'_{3R}, E'_{3G}, E'_{3B}, E'_{4R}, E'_{4G}, E'_{4B})$ are given as the input to the NN. The random color reflectance factors $(\rho'_R, \rho'_G, \rho'_B)$ are given as the output data. After learning of NN, the input data $(E_{1R}, E_{1G}, E_{1B}, E_{2R}, E_{2G}, E_{2B}, E_{3R}, E_{3G}, E_{3B}, E_{4R}, E_{4G}, E_{4B})$ of a target object are given, then, the color reflectance factor (ρ_R, ρ_G, ρ_B) of the target object is obtained from NN as the output data in the generalization of NN. The architecture is shown in Fig. 5.

3.3. NN Implementation for Shape Recovery

Selected feature points can construct a virtual sphere from the target object itself. Then the mapping of (E_1, E_2, E_3, E_4) to (n_x, n_y, n_z) of the selected feature points is used as the learning data under four light sources.

When NN based monochrome photometric stereo is expanded to color object to recover the shape, the number of learning data increases. Although the number of color data is 65536 times more than the monochrome data, it is necessary to prepare the learning data efficiently. Further,

(E_R, E_G, E_B) is synthesized by random color reflectance factor ρ' for each point on the learning data to cover the color of all points on the target object, the following relation $(E'_R, E'_G, E'_B) = (\rho'_R E_R, \rho'_G E_G, \rho'_B E_B)$ are used. The diffuse component and the specular component are separated in Eq. (3). That is, only the diffuse component is synthesized and added for each point on the learning data by ρ' .

NN learning with monochrome data generally gives the higher accuracy than that with the original color data because of the dimensionality problem. This means that one channel data with the largest dynamic range is used to learn the NN mapping for the higher accuracy. Therefore, the largest range data among (E'_R, E'_G, E'_B) selected for each pixel included in the target object are used as the learning data E' .

$$E' = \max(E'_R, E'_G, E'_B). \quad \dots \dots \dots (10)$$

In the learning during self-calibration, the learning data obtained from the feature points are prepared using Eq. (10) and (E'_1, E'_2, E'_3, E'_4) are given as the input to NN. The corresponding (n_x, n_y, n_z) are given as the output of NN. After the learning of NN, input data (E_1, E_2, E_3, E_4) of a target object are given, then, the corresponding (n_x, n_y, n_z) of the target object is obtained through the NN as output. The architecture for the generalization of this RBF-NN (Radial Basis Function Neural Network) is shown in Fig. 6.

4. Neural Network Based Rendering

Given the surface orientation and the color reflectance factor, a virtual image can be rendered for any viewpoint under any direction of the light source. The rendered image intensity E can be represented using the incident angle i , the emittance angle e , and the phase angle g with the color reflectance factor ρ .

The previous approach of neural network based rendering [6] uses (i, e, g) shown in Eq. (11).

$$\begin{aligned} i &= \cos^{-1}(\mathbf{n} \cdot \mathbf{s}) \\ e &= \cos^{-1}(\mathbf{n} \cdot \mathbf{v}) \\ g &= \cos^{-1}(\mathbf{v} \cdot \mathbf{s}) \quad \dots \dots \dots (11) \end{aligned}$$

Here, instead of using the phase angle g , the angle h is defined in Eq. (12).

$$h = \cos^{-1}(\mathbf{d} \cdot \mathbf{n}) \quad \dots \dots \dots (12)$$

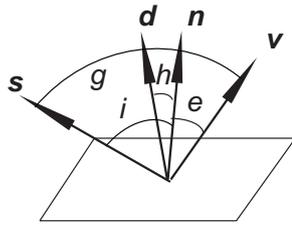


Fig. 7. (i, e, h) .

where h is the angle between the surface normal vector \mathbf{n} and a vector \mathbf{d} , and the vector \mathbf{d} is equally divided vector between the light source direction vector \mathbf{s} and the viewing direction vector \mathbf{v} as shown in Fig. 7. In the proposed approach, E is derived from Eq. (13). Here, R_m and E_m means functions of $(\mathbf{n}, \mathbf{s}, \mathbf{v})$ and (i, e, h) , respectively.

$$E = R_d(\mathbf{n}, \mathbf{s})\rho + R_m(\mathbf{n}, \mathbf{s}, \mathbf{v}) = E_d(i)\rho + E_m(i, e, h) \quad (13)$$

Here, the range of (i, e, h) is given as

$$0^\circ \leq i \leq 90^\circ, \quad 0^\circ \leq e \leq 90^\circ, \quad 0^\circ \leq h \leq 90^\circ. \quad (14)$$

When we use four light sources, the variation of g is restricted to only four values. Sparse sampling results in the lower accuracy for generating image in general, as shown in Fig. 8. It is necessary to use a large number of images to obtain the high accuracy for the rendering. That is, a large number of light sources are required when the neural network is used to learn data from (i, e, g) space.

From the comparison between (i, e, g) space shown in Fig. 8 and (i, e, h) space shown in Fig. 9, it is shown that (i, e, h) space builds data sampling with few gaps. Using (i, e, h) space can improve the accuracy of the neural network learning.

The mapping of (i, e, h) to E_m is learned for the rendering of the specular components of the target object. After the learning, E_m is generalized using the rendering NN. The architecture for the generalization of this RBF-NN is shown in Fig. 10. Here, E_d can be calculated by $\cos i = \mathbf{n} \cdot \mathbf{s}$ and E_m is estimated using an RBF neural network. The rendered image intensity (E_R, E_G, E_B) can be separately applied to Eq. (13).

5. Experimental Results

Figure 11 illustrates the observation environment. Four light sources are used to illuminate the target object. Let \mathbf{v} be $(0,0,1)$, and it is necessary to estimate \mathbf{s} for the neural network based rendering, one light source direction is aligned with the camera viewing direction to check whether the feature point extracted is good candidate or not. The (f, g) of each feature point also is tracked as a function of the rotation angle α . When the target object is rotated from 0° to 180° , each feature point extracted from the occluding boundary is tracked horizontally during rotation and the corresponding image intensities E are observed. For the candidate of feature point observed under

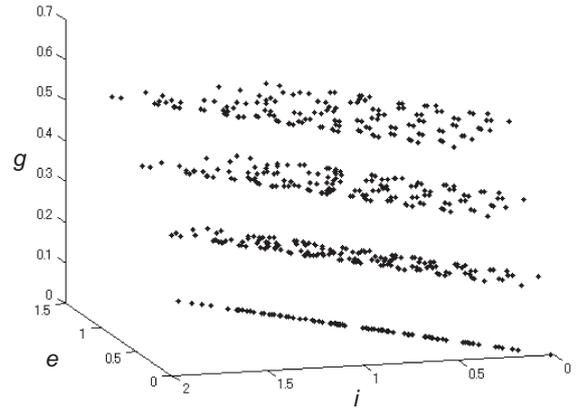


Fig. 8. Sampling data in (i, e, g) space.

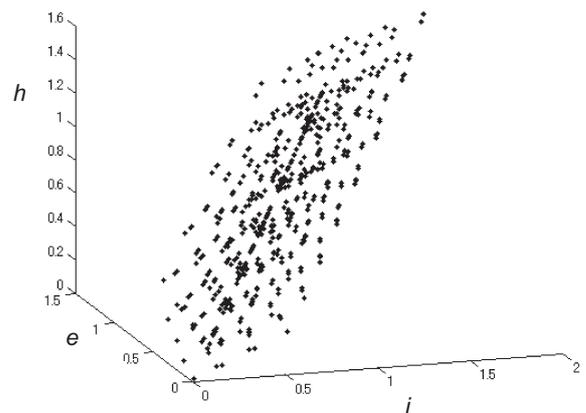


Fig. 9. Sampling data in (i, e, h) space.

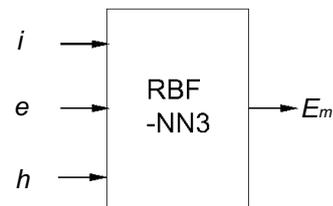


Fig. 10. NN rendering for specular component.

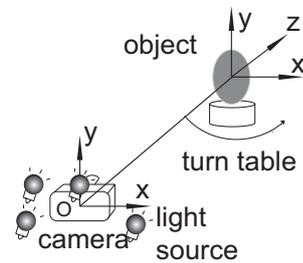
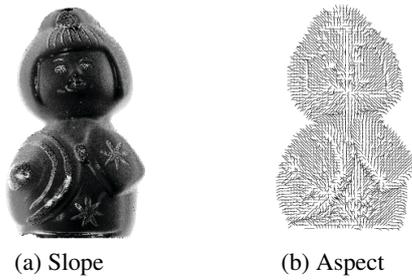


Fig. 11. Observation environment.

the light source direction which is the same as the viewing direction, E should be increased from 0° to 90° rotation, while E should be decreased from 90° to 180° rotation. Only the data of feature points which satisfy this condi-



(a) Slope (b) Aspect
Fig. 12. Recovered surface shape.



(a) Actual input image (b) Color reflectance
Fig. 13. Recovered surface shape.



(a) Any light source direction 1 (b) Any light source direction 2
Fig. 14. Results of virtual image rendering.

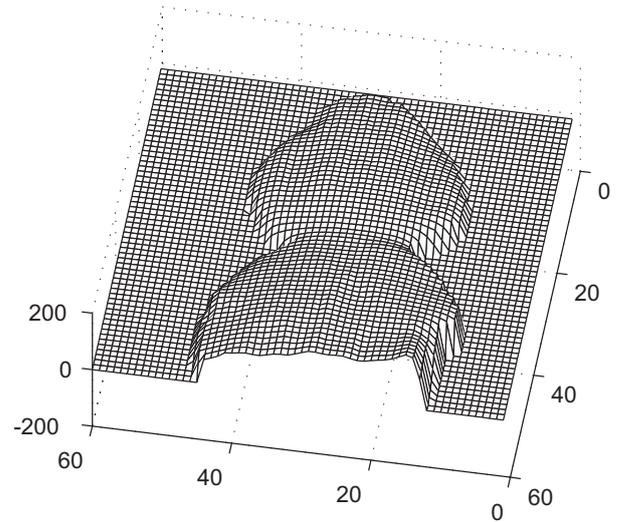


Fig. 15. Height distribution.

object using self-calibration and NN are shown in **Fig. 12** and **Fig. 13**. Here, surface orientation (surface gradient) of each point on the target object is represented as both slope and aspect. Slope represents the steepness of surface gradient and aspect represents the azimuth of gradient. **Fig. 12(a)** linearly encodes the slope angle e (i.e., $\tan^{-1}(\sqrt{p^2 + q^2})$, the angle between the surface normal and the viewing direction) as a gray value in the range of black ($e = 0$) to white ($e = \pi/2$), while **Fig. 12(b)** plots the aspect angle (i.e., $\tan^{-1}(q/p)$, the projection of the surface normal onto the XY -plane) as a short line segment. **Fig. 13(b)** encodes the color albedo (color reflectance factor). Both the surface orientation and color reflectance factor are recovered by the proposed self-calibration approach without using any calibration sphere.

The virtual image means the generated image by the proposed method for any viewpoint and any direction of light source after obtaining 3-D model and color reflectance factor acquired by each NN. The virtual image under any light source direction is shown in **Figs. 14(a)** and **(b)**. The light source direction is given as the vector $(0.6635, -0.0084, 0.7656)$ in **Fig. 14(a)**. The light source direction is given as the vector $(-0.6622, 0.0903, 0.7438)$ in **Fig. 14(b)**. It is shown that the virtual image rendering gives the realistic feelings for both of them.

The height distribution obtained by the integration of surface orientation is shown in **Fig. 15**. Rotating this height distribution and the rendering NN can generate a realistic virtual image at any viewpoint. The results are shown in **Figs. 16(a)** and **(b)** from the different views.

From these results, the proposed approach can generate the virtual images of the target object for any viewpoint and any direction of light source. In the experiments, the multiple color object in **Fig. 17(a)** is used to obtain results, and error analysis is performed on the target object.

Figure 17(a) shows the actual image in the light source direction $(0.6635, -0.0084, 0.7656)$. Using the simpler

tion are used for the NN learning. The learning data are sorted with the unique combinations of (f, g) and input to NN. For the details, see [10].

Four images are obtained under four different conditions of illumination for each object pose during rotation. The target object is rotated with every 1 degree between 0° to 359° degrees. A total of 360×4 images are taken and used in this experiment. Actual input image consists of 8-bit for each of RGB, and this input image intensity E is normalized to the real value between 0 and 1. The range of i, e, h takes 0 to $\pi/2$. The spread constant of RBF-NN takes a half of the maximum of the input data in the NN learning. In the NN learning for shape recovery and color reflectance factor, the number of learning data set is 4029, the number of learning epochs is 50, and the spread constant is 0.5. In the NN based rendering, the number of learning set is 5644, and the spread constant is 0.8.

The higher density of data is used in the neural network, the better result is obtained in the glossy points. This means that dense data are required when the neural network uses the learning data in the glossy points. This experiment is performed so that the target object is rotated with every 1° step to obtain the dense observed data.

The recovered shape and color reflectance of the target



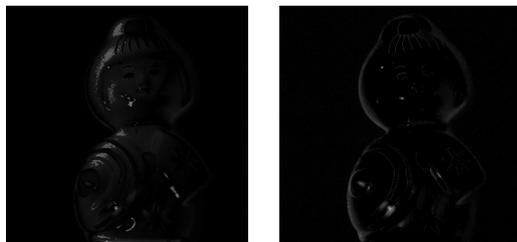
(a) Any view point 1 (b) Any view point 2
Fig. 16. Results of virtual image rendering.



(a) Actual input image (b) Virtual image by Torrance-Sparrow model
Fig. 17. Results of virtual image rendering.



(a) Virtual image rendering using (i, e, g) space (b) Virtual image rendering using (i, e, h) space
Fig. 18. Difference between real virtual light image.



(a) (i, e, g) space (b) (i, e, h) space
Fig. 19. Difference between real virtual light image.

formula of Torrance-Sparrow model $E_m = k e^{\frac{-h^2}{2\sigma^2 \cos i}}$ [11], the parameter $k = 0.5$ and $\sigma = 0.01$, **Fig. 17(b)** is the virtual image by Torrance-Sparrow model in the same light source direction. The mean error between actual image **Fig. 17(a)** and the virtual image by Torrance-Sparrow model **Fig. 17(b)** is 7.9520, and the variance is 0.0061. **Fig. 18(a)** shows the result in the same light source direction using the (i, e, g) space. **Fig. 18(b)** shows the result in the same light source direction using the (i, e, h) space. The difference between actual image **Fig. 17(a)** and virtual image **Fig. 18(a)** is shown in **Fig. 19(a)**. The mean error between actual image **Fig. 17(a)** and virtual image **Fig. 18(a)** is 10.5645, and the variance is 0.0167. The dif-

Table 1. Mean and variance error.

	mean error	variance of error
(i, e, g) space	10.5645	0.0167
(i, e, h) space	3.3623	0.0031
Torrance-Sparrow model	7.9520	0.0061

ference between actual image **Fig. 17(a)** and virtual image **Fig. 18(b)** is shown in **Fig. 19(b)**. The mean error of real image **Fig. 17(a)** and virtual image **Fig. 18(b)** is 3.3623, and the variance is 0.0031. The result is shown in **Table 1**. Here, it is confirmed that the result obtained by (i, e, h) space is better than that by (i, e, g) space from the rendering results including the glossy points. From the comparison between the virtual image by Torrance-Sparrow model and that by the proposed method, the proposed method using (i, e, h) space with NN based rendering is better than the method by model based rendering.

6. Conclusion

This paper proposed a new method of self-calibration and color image rendering using RBF-NN without using any calibration object. The learning data is obtained by rotating the target object itself without assuming any functional model of the reflectance function. With four input images, both the surface orientation and color reflectance factor are obtained using NN from the target object with multiple color reflectance factors. Further, a virtual image under any viewpoint and any direction of light source can be obtained with rendering NN. It is shown that the proposed approach is effective through the experimental results for the real object. Cast shadow is another problem but this remains as the future work.

Acknowledgements

The authors thank lab. colleagues at NIT and Chubu Univ. for useful discussions. Ding's research is supported by the Hori Information Science Promotion Foundation. Iwahori's research is supported by JSPS Grant-in-Aid for Scientific Research (C) (2050 0168) and Chubu Univ. Grant. Woodham's research is supported by NSERC.

References:

- [1] R. J. Woodham, "Photometric Method for Determining Surface Orientation from Multiple Images," *Opt. Engineering*, pp. 139-144, 1980.
- [2] R. J. Woodham, "Gradient and Curvature from the Photometric-Stereo Method, Including Local Confidence Estimation," *J. Opt. Soc. Am., A*, pp. 3050-3068, 1994.
- [3] Y. Iwahori, R. J. Woodham, and A. Bagheri, "Principal Components Analysis and Neural Network Implementation of Photometric Stereo," *Proc. IEEE Workshop on Physics-Based Modeling in Computer Vision*, pp. 117-125, 1995.
- [4] S. Chen, Y. Wu, and B. L. Luk, "Combined Genetic Algorithm Optimization and Regularized Orthogonal Least Squares Learning for Radial Basis Function Networks," *IEEE Trans. on Neural Networks*, Vol.10, No.5, pp. 1239-1243, 1999. ISSN 1045-9227.

- [5] Y. Iwahori, R. J. Woodham, S. Bhuiyan, and N. Ishii, "Neural Network Based Photometric Stereo for Object with Non-Uniform Reflectance Factor," Proc. of 6th Int. Conf. on Neural Information Processing (ICONIP'99), Vol.III, pp. 1213-1218, 1999.
- [6] H. Kawanaka, Y. Iwahori, R. J. Woodham, and K. Funahashi, "Color Photometric Stereo and Virtual Image Rendering Using Neural Network," Trans. of IEICE, Vol.J89-D-II, No.2, pp. 381-392, 2006. (in Japanese)
- [7] K. Ikeuchi, Y. Sato, K. Nishino, and I. Sato, "Modeling from Reality: Photometric Aspect," Trans. of Virtual Reality Society in Japan, Vol.4, No.4, pp. 623-630, 1999. (in Japanese)
- [8] A. Laurentini, "How Far 3D Shapes Can Be Understood from 2D Silhouettes," IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol.17, No.2, pp. 188-195, 1995.
- [9] S. Hiura, K. Sato, and S. Inokuchi, "Simultaneous Measurement of Shape and Reflectance by Rotating Object," Trans. of Information Processing Society of Japan, Vol.36, No.10, pp. 2295-2302, 1995.
- [10] Y. Iwahori, Y. Watanabe, K. Funahashi, and R. J. Woodham, "Self-Calibrated Neural Network Based Photometric Stereo," Int. J. of Computer and Information Science, Vol.3, No.1, 40359, 2002.
- [11] G. Meister, R. Wiemker, R. Monno, H. Spitzer, and A. Strahler, "Investigation on the Torrance-Sparrow Specular BRDF Model," IGARSS '98, Vol.4, pp. 2095-2097, 1998.



Name:
Yuji Iwahori

Affiliation:
Dept. of Computer Science, Chubu University

Address:
1200 Matsumoto-cho, Kasugai 487-8501, Japan

Brief Biographical History:
1988 Ph.D. in Engineering, Tokyo Institute of Technology, Japan
1988- Research Associate, Educational Center for Information Processing, Nagoya Institute of Technology, Japan
1992- Associate Professor, Nagoya Institute of Technology, Japan
2002- Professor, Nagoya Institute of Technology, Japan
2004- Professor, Dept. of Computer Science, Chubu University, Japan (Visiting Professor, Dept. of Comp. Sci., UBC, Canada)

Main Works:
• "Neural Network based Photometric Stereo with a Nearby Rotational Moving Light Source," IEICE Trans. on Information and Systems, Vol.E80-D, No.9, pp. 948-957, 1997.
• "Obtaining Shape from Scanning Electron Microscope using Hopfield Neural Network," J. of Intelligent Manufacturing, Vol.16, No.6, pp. 715-725, Dec. 2005.

Membership in Academic Societies:
• The IEEE Computer Society (IEEE-CS)
• The Institute of Electronics, Information and Communication Engineers (IEICE)
• The Information Processing Society of Japan (IPSI)
• The Knowledge-Based and Intelligent Information & Engineering Systems (KES)



Name:
Yi Ding

Affiliation:
Department of Artificial Intelligence and Computer Science, Nagoya Institute of Technology

Address:
Gokiso-cho, Showa-ku, Nagoya 466-8555, Japan

Brief Biographical History:
2007 M.S. degree, Nagoya Institute of Technology
2007- Ph.D. Course Graduate Student, Nagoya Institute of Technology

Main Works:
• "Relative Magnitude of Gaussian Curvature Using Neural Network and Object Rotation of Two Degrees of Freedom," MVA2007, Vol.8, No.20, pp. 307-310, 2007.

Membership in Academic Societies:
• The Information Processing Society of Japan (IPSI)



Name:
Tsuyoshi Nakamura

Affiliation:
Department of Computer Science and Engineering, Nagoya Institute of Technology

Address:
Gokiso-cho, Showa-ku, Nagoya 466-8555, Japan

Brief Biographical History:
1998 Ph.D., Nagoya Institute of Technology
1998-2003 Research associate, Nagoya Institute of Technology
2003- Associate professor, Nagoya Institute of Technology

Main Works:
• "Rough Set Based FCM Algorithm for Image Segmentation," Int. J. of Computational Science, Vol.1 No.1, pp. 58-68, 2007.

Membership in Academic Societies:
• The Institute of Electrical and Electronics Engineering (IEEE)
• Association for Computing Machinery (ACM)
• The Institute of Electronics, Information and Communication Engineers (IEICE)
• Japan Society for Fuzzy Theory and Intelligent Informatics (SOFT)
• Japan Society of Kansei Engineering (JSKE)



Name:
Lifeng He

Affiliation:
Faculty of Information Science and Technology,
Aichi Prefectural University

Address:

Kumabari-Ibaragabasama, Nagakute, Aichi-gun, Aichi 480-1198, Japan

Brief Biographical History:

1997 Ph.D., Nagoya Institute of Technology
1997-1999 Research Associate, Aichi Prefectural University
1999- Associate Professor, Aichi Prefectural University

Main Works:

- "A Run-Based Two-Scan Labeling Algorithm," IEEE Trans. on Image Processing, Vol.17, No.5, pp. 749-756, 2008.

Membership in Academic Societies:

- Associate of Automated Reasoning (AAR)
-



Name:
Hidenori Itoh

Affiliation:
Department of Computer Science and Engineering,
Nagoya Institute of Technology

Address:

Gokiso-cho, Showa-ku, Nagoya 466-8555, Japan

Brief Biographical History:

1974 Ph.D., Nagoya University
1989- Professor, Nagoya Institute of Technology

Main Works:

- "Rough Set Based FCM Algorithm for Image Segmentation," Int. J. of Computational Science, Vol.1 No.1, pp. 58-68, 2007.

Membership in Academic Societies:

- The Institute of Electronics, Information and Communication Engineers (IEICE)
 - Japan Society of Kansei Engineering (JSKE)
-



Name:
Robert J. Woodham

Affiliation:
University of British Columbia

Address:

Vancouver, B.C. V6T 1Z4, Canada

Brief Biographical History:

1977 Ph.D., Massachusetts Institute of Technology
1990- Professor, University of British Columbia

Main Works:

- "Gradient and Curvature from the Photometric Stereo Method, Including Local Confidence Estimation," J. of the Optical Society of America, A, Vol.11, pp. 3050-3068, 1994.

Membership in Academic Societies:

- The IEEE Computer Society (IEEE-CS)
-