

author: Jason Jeong (zljieong)  
title: 4901L assignment 4

### 1.1 Lambertian albedo

### 1.2 Foreshortening

1.2.1 Calculate the solid angle subtended by  $dA$  at points  $X_1$  and  $X_2$ .

1.2.2 Calculate the irradiance  $E$  incident on the plane at points  $X_1$  and  $X_2$ , and calculate the ratio  $E(X_1)/E(X_2)$ .

### 1.3 Simple rendering

1.3.1 Use quiver to display the x-y components of the normal field and print the result.

1.3.2 • Compute and display the radiance emitted from each point assuming Lambertian reflectance and constant albedo (equal to one), with a distant point light source in direction  $\hat{s} = (0, 0, 1)$ , by typing `imshow(N(:, :, 3))`. Explain why this works.

Compute, display and print the emitted radiance for three different light source directions which are rotated i)  $45^\circ$  up, ii)  $45^\circ$  right, and iii)  $75^\circ$  right from the frontal direction  $\hat{s} = (0, 0, 1)$ . Can you spot errors in the field of surface normals? What are the illumination effects being ignored in this calculation of scene radiance?

*i)  $45^\circ$  up*

*ii)  $45^\circ$  right*

*ii)  $75^\circ$  right*

### 1.4 Photometric stereo

1.4.1 Estimate the surface normal and gray scale albedo for each pixel. Submit your Matlab code as well as the results (using `imshow` for the albedo values and `quiver` for the surface normals.)

1.4.2 Note the poor estimates of the albedo (and surface normal) in the area surrounding the nostrils. What is the source of this error?

Describe one method for finding a better estimate of this information from these seven images.

1.4.3 Use the recovered surface information to predict what the person would look like (in grayscale) if illuminated from direction  $\hat{s} = (0.58, -0.58, -0.58)$  and from direction  $\hat{s} = (-0.58, -0.58, -0.58)$ . Submit your results and your code.

The function `integrate frankot.m` in the matlab folder can be used to recover a surface  $z(x, y)$  from your surface normals  $\hat{n}(x, y)$ , and the surface can be displayed in Matlab. Display and submit two views of the recovered shape.

### 1.5 Dichromatic reflectance

1.5.1 Assuming that the spectral sensitivities of a camera's three filters are  $(c_R(\lambda), c_G(\lambda), c_B(\lambda))$  and that the BRDF at the surface point imaged at pixel  $\vec{u}$  is  $f(\lambda, \hat{\omega}_i(\vec{u}), \hat{\omega}_o(\vec{u})) = f_d(\lambda, \vec{u}) + f_s(\hat{\omega}_i(\vec{u}), \hat{\omega}_o(\vec{u}))$ , write expressions for the elements of the diffuse color vectors  $\vec{d}(\vec{u})$  and the source color vector  $f_s$ .

1.5.2 Suppose you are given two unit-length three-vectors  $\hat{r}_1$  and  $\hat{r}_2$  that are orthogonal to  $\vec{s}$ . Show that the two-channel image1 given by the per-pixel inner products ( $\langle \hat{r}_1, \hat{C}(\hat{u}) \rangle, \langle \hat{r}_2, \hat{C}(\hat{u}) \rangle$ ):

a) does not depend on the specular components of the BRDFs,

$$f_s(\hat{w}_i(\vec{u}), \hat{w}_o(\vec{u}))$$

b) depends linearly on the surface normals,  $\hat{n}(\vec{u})$ .

1.5.3 Show that the two properties from part (b) are also satisfied by the single-channel (gray scale) image  $J(\vec{u}) = ||J(\vec{u})||$ .

1.5.4 Write a function `imout=makeLambertian(im,s)` that takes an RGB image `im` and a source color vector `s` and computes the grayscale image  $J(\sim u)$  from part (c). Run this function on the image `fruitbowl.png` from the data folder using `s=[0.6257 0.5678 0.5349]` and submit the results. Explain in three sentences or less why `imout` might be more useful than `im` to a computer vision system. Note that a simple way to numerically obtain a set of  $N - 1$  unit-length vectors that are orthogonal to  $N$ -vector is to use the `null` command in Matlab.

1.6 Color metamers (Extra Credit)

1.6.1 Show that the tristimulus vector  $\vec{C}_f = (X_f, Y_f, Z_f)$  for a ripe banana  $f(\lambda)$  under illuminant spectrum  $l(\lambda)$  can be written as  $\vec{C}_f = L_f \vec{l}$  where  $L_f$  is a  $3 \times N$  matrix and  $\vec{l}$  is the discretization of  $l(\lambda)$ .

1.6.2 Given  $\vec{f}$  and  $\vec{g}$ , a good way to choose  $\vec{l}$  is to minimize the distance, in a least-squares sense, of the two resulting tristimulus vectors. Write an expression for the Euclidean distance between the two tristimulus vectors in terms of  $L_f$ , its counterpart  $L_g$ , and  $\vec{l}$ .

1.6.4 The file `bananas.mat` from the data folder contains two spectral reflectance functions, `ripe` and `overripe`, which are sampled in 10nm increments from 400nm to 700nm and correspond roughly to the data from Day 5 and Day 7 of Figure 4. Use your function to find the optimal metamer-inducing blackbody temperature for these materials, and plot the distance as a function of temperature, and the spectral power distribution of the best illuminant. You will need to use the file `ciexyz64 1.csv` from the data folder, which contains tabulated data for the CIE XYZ color matching functions. In Matlab, you can read data from a CSV file using the `csvread` command. Remember to normalize the spectral power distributions of blackbody radiators in a way that is meaningful for the minimization criterion of part (b).

## 1.1 Lambertian albedo

The energy coming in is

$$\int_{\Omega} L_{src}(\theta_{in}, \phi_{in}) \cos \theta_{in} dA d\Omega$$

and the energy leaving the surface is

$$\int_{\Omega} L_{surface}(\theta_{out}, \phi_{out}) \cos \theta_{out} dA d\Omega$$

$$= \int_{\Omega} \int_{\Omega_{in}} f(\theta_{in}, \phi_{in}, \theta_{out}, \phi_{out}) L_{src}(\theta_{out}, \phi_{out}) \cos \theta_{in} dA d\Omega_{in} \cos \theta_{out} dA d\Omega$$

$$= \int_{\Omega} \int_{\Omega_{in}} \frac{\rho}{\pi} L_{src}(\theta_{out}, \phi_{out}) \cos \theta_{in} d\Omega_{in} \cos \theta_{out} d\Omega dA$$

$$\begin{aligned}
&= \frac{\rho}{\pi} \int_{\Omega_{in}} L_{src}(\theta_{out}, \phi_{out}) \cos \theta_{in} dA d\Omega_{in} \int_{\Omega} \cos \theta_{out} dA d\Omega \\
&= \frac{\rho}{\pi} \int_{\Omega_{in}} L_{src}(\theta_{out}, \phi_{out}) \cos \theta_{in} dA d\Omega_{in} \int_0^{2\pi} \int_0^{\frac{\pi}{2}} \cos \theta \sin \theta dA d\theta d\phi \\
&= \frac{\rho}{\pi} \int_{\Omega_{in}} L_{src}(\theta_{out}, \phi_{out}) \cos \theta_{in} dA d\Omega_{in} \cdot \pi \\
&= \rho \int_{\Omega_{in}} L_{src}(\theta_{out}, \phi_{out}) \cos \theta_{in} dA d\Omega_{in}
\end{aligned}$$

By the conversion of energy, clearly the energy leaving is smaller or equal to the energy coming, then:

$$\int_{\Omega} L_{src}(\theta_{in}, \phi_{in}) \cos \theta_{in} d\Omega dA \geq \rho \int_{\Omega_{in}} L_{src}(\theta_{out}, \phi_{out}) \cos \theta_{in} dA d\Omega_{in}$$

$$1 \geq \rho$$

known that both sides are energy which cannot be negative, we have:

$$0 \leq \rho \leq 1$$

## 1.2 Foreshortening

### 1.2.1 Calculate the solid angle subtended by dA at points X1 and X2.

$$d\Omega_{X_1} = \frac{dA \cos 0}{D^2} = \frac{dA}{D^2}$$

$$d\Omega_{X_2} = \frac{dA \cos \alpha}{\left(\frac{D}{\cos \alpha}\right)^2} = \frac{dA \cos^3 \alpha}{D^2}$$

### 1.2.2 Calculate the irradiance E incident on the plane at points X\_1 and X\_2, and calculate the ratio $E(X_1)/E(X_2)$ .

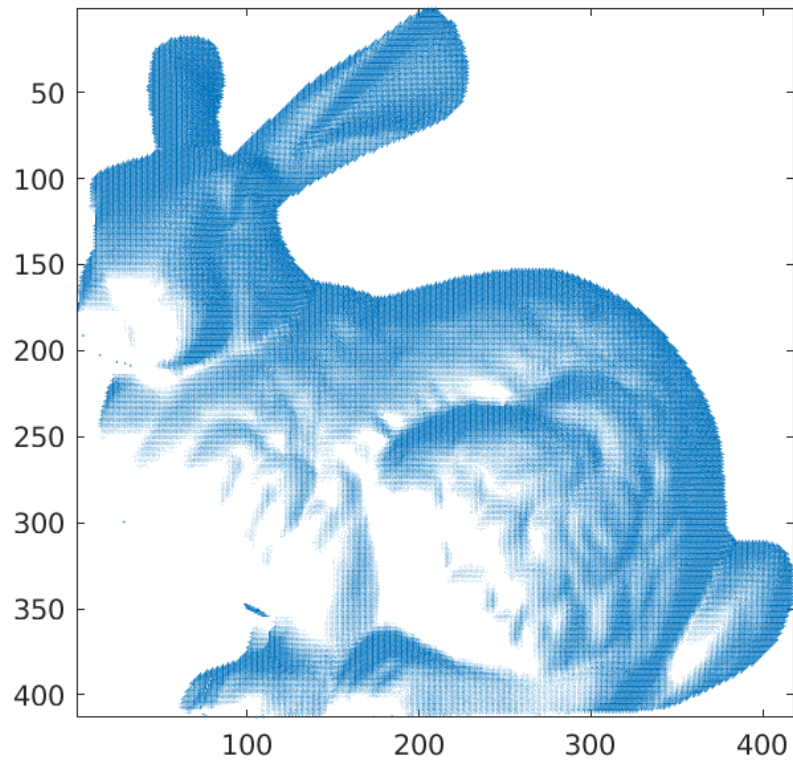
$$E(X_1) = L \cos 0 d\Omega_{X_1} = \frac{L dA}{D^2}$$

$$E(X_2) = L \cos \alpha d\Omega_{X_2} = \frac{L \cos^4(\alpha) dA}{D^2}$$

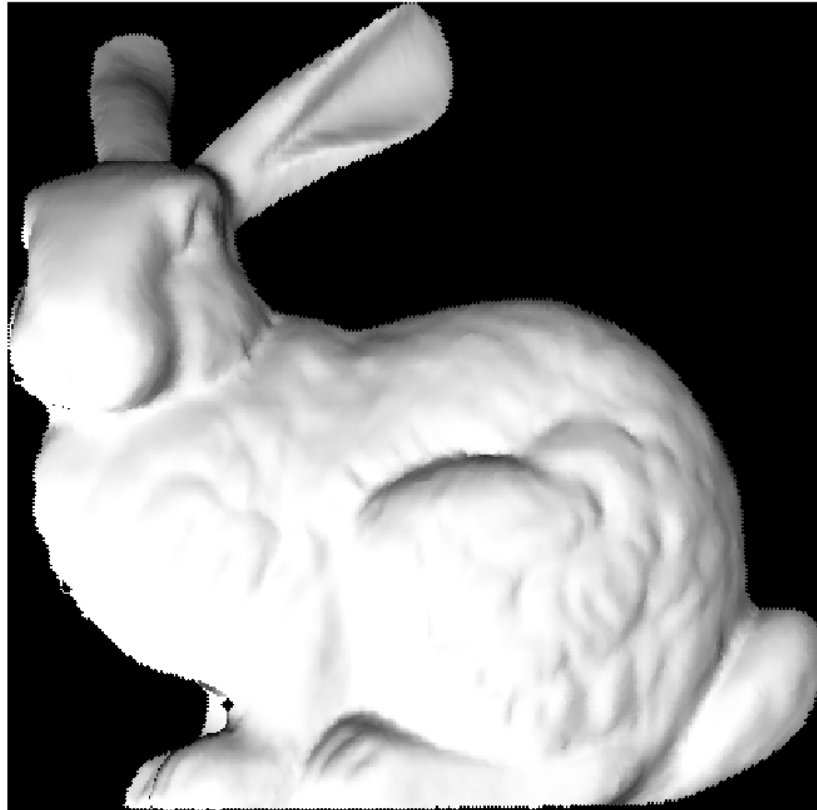
$$\frac{E(X_1)}{E(X_2)} = \frac{1}{\cos^4 \alpha}$$

## 1.3 Simple rendering

### 1.3.1 Use quiver to display the x-y components of the normal field and print the result.



**1.3.2 • Compute and display the radiance emitted from each point assuming Lambertian reflectance and constant albedo (equal to one), with a distant point light source in direction  $\hat{s} = (0, 0, 1)$ , by typing `imshow(N(:,:,3))`. Explain why this works.**



Why this work:

Under the assumption that all source of incoming flux is from one direction,

$$L(x, \omega) \longrightarrow L(\omega) \longrightarrow s\delta(\omega = \omega_o)$$

Assume Lambertian and constant BRDF, the radiance emitted due to reflectance at  $\omega_{out}$

$$= \rho \int_{\Omega} s\delta(\omega = \omega_{out}) \hat{\omega} \cdot \hat{n} \, d\Omega$$

$$= \rho s \cdot \hat{n} \cdot \hat{w}_i$$

$$= \rho \hat{n} \cdot \vec{s}$$

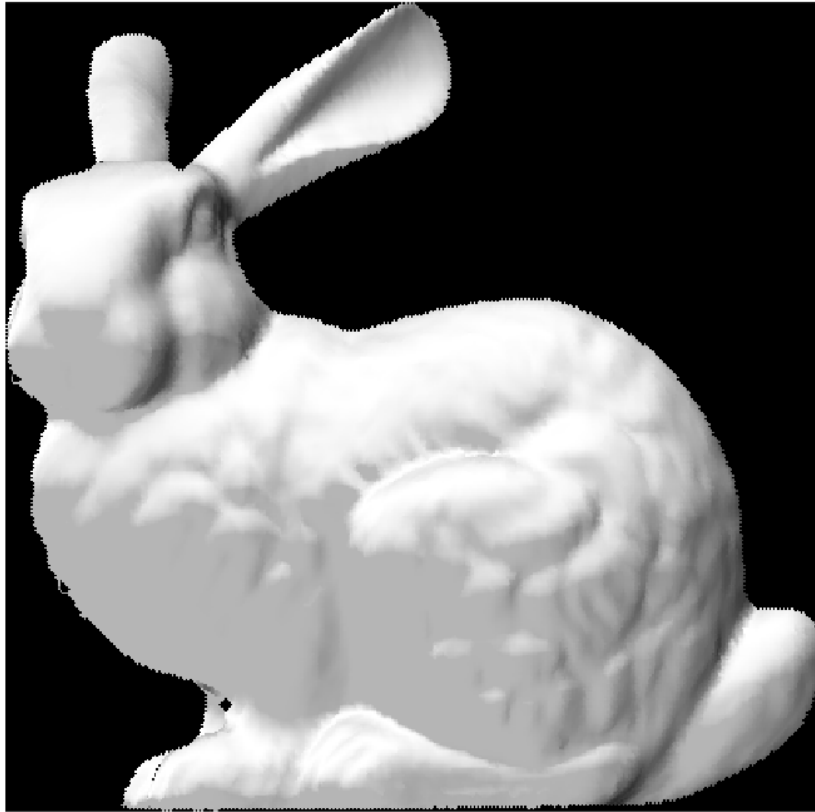
Where  $\rho$  is assumed to be 1, then  $\vec{s} = \hat{w}_i$

$L(\hat{w}_{out}) = \hat{n} \cdot (0, 0, 1)$  gives the result of source from  $(0, 0, 1)$

**Compute, display and print the emitted radiance for three different light source directions which are rotated i) 45° up, ii) 45° right, and iii) 75° right from the frontal direction  $\hat{s} = (0, 0, 1)$ . Can you spot errors in the field of surface normals? What are the illumination effects being ignored in this calculation of scene radiance?**

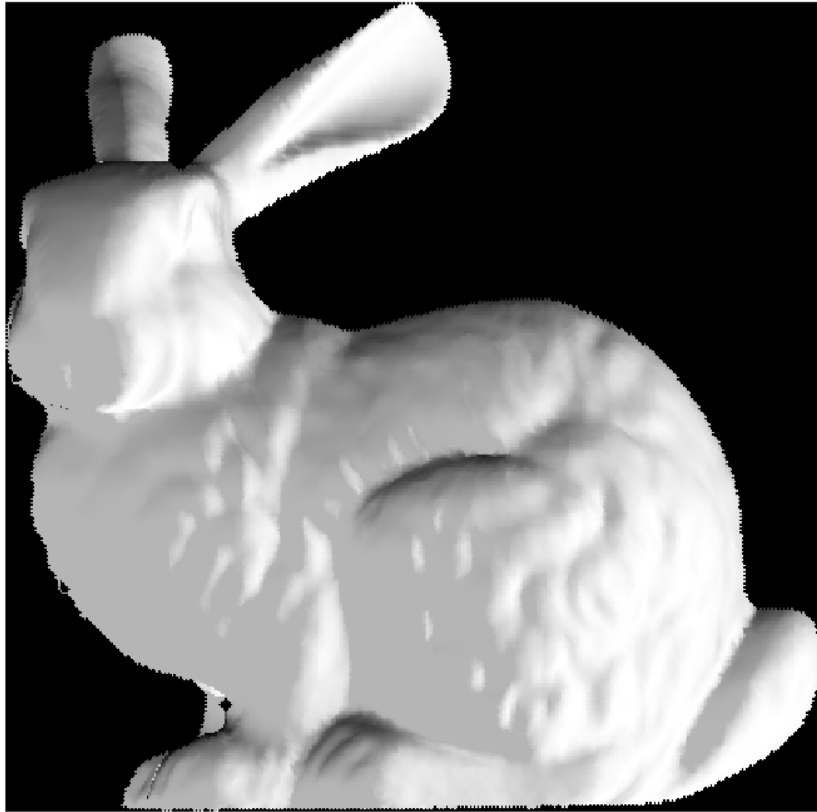
i) 45° up

$$\hat{w}_i = (0, \sin 45^\circ, \cos 45^\circ)$$



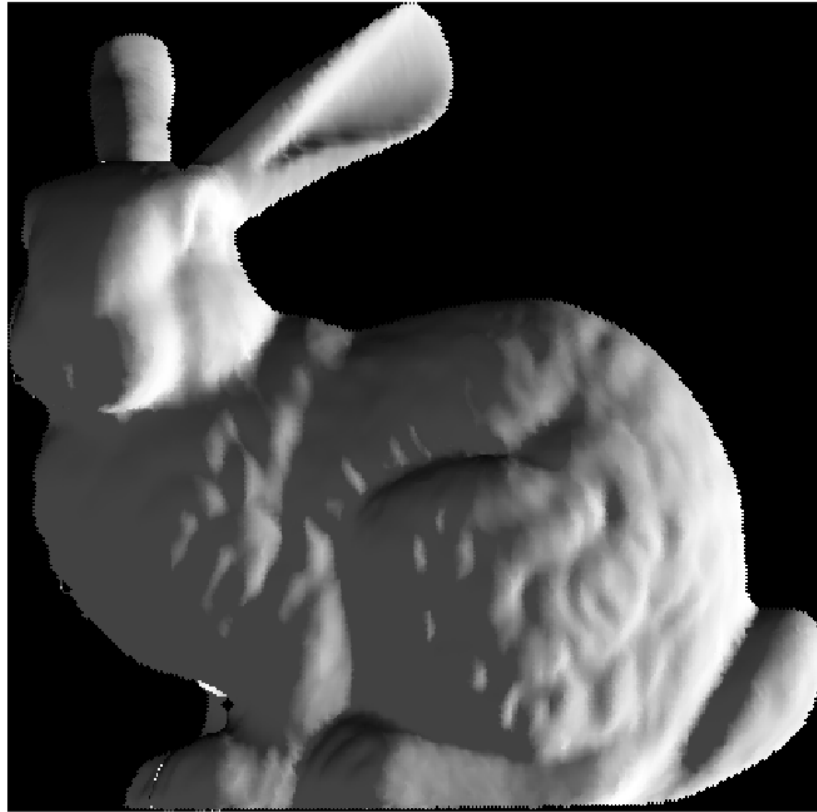
*ii)  $45^\circ$  right*

$$\hat{w}_i = (\sin 45^\circ, 0, \cos 45^\circ)$$



*ii)  $75^\circ$  right*

$$\hat{w}_i = (\sin 75^\circ, 0, \cos 75^\circ)$$



Error: above the foot and under the ears.

Illumination effects being ignored: shadows, the actual albedo, and the actual non Lambertian surface.

## 1.4 Photometric stereo

**1.4.1 Estimate the surface normal and gray scale albedo for each pixel. Submit your Matlab code as well as the results (using imshow for the albedo values and quiver for the surface normals.)**

using the result from 1.3, we have

$$I = \rho \cdot \hat{n} \cdot \vec{s} = \vec{s}^T * (\rho \cdot \hat{n})$$

Then let  $\vec{g} = \rho \cdot \hat{n}$ , with multiple source and output images, we may solve  $\rho$  and  $\vec{n}$  by least squares Approximations

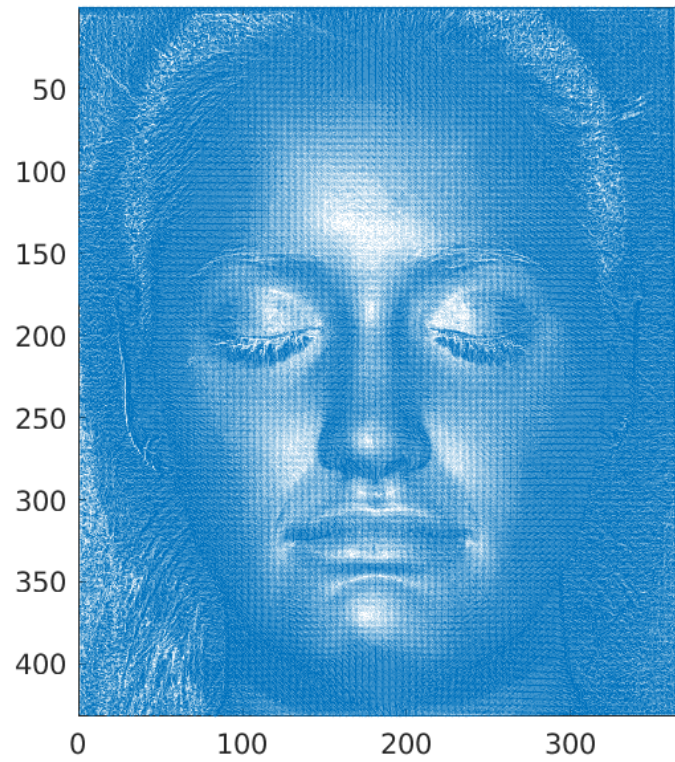
$$I = (\vec{s}_0 \ \vec{s}_1 \ \dots \ \vec{s}_n) * \vec{g} = S\vec{g}$$

$$\vec{g} = (S^T S)^{-1} S^T I$$

where since  $\rho$  are scalar and  $\hat{n}$  are unit vector and can be decomposed after getting  $\vec{g}$



Matlab code: in s1\_4.m  
albedo:



normal:



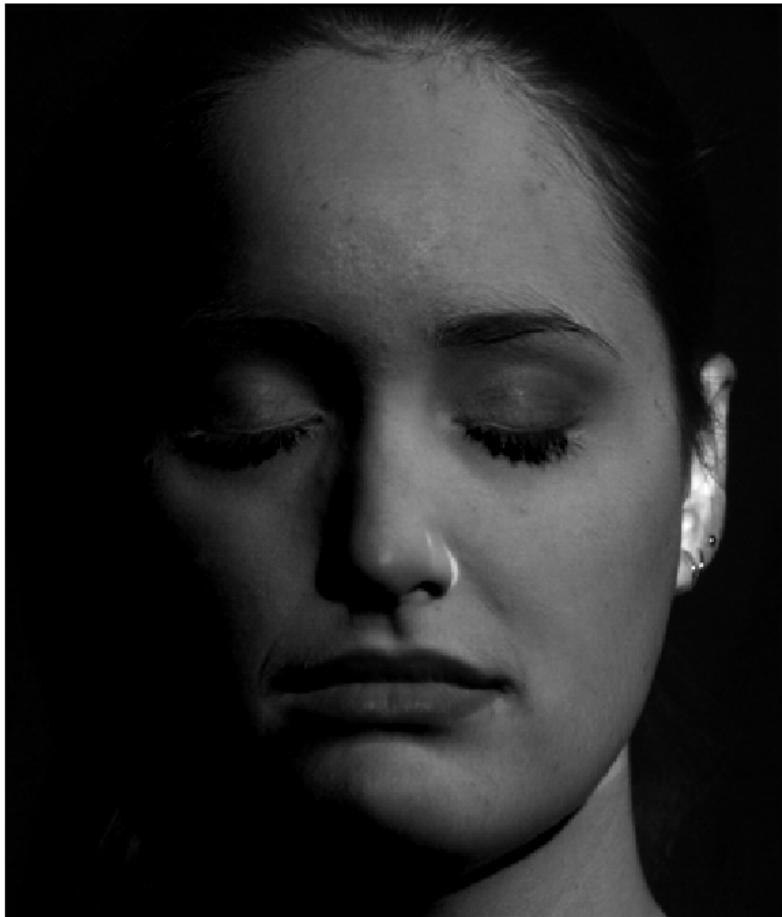
**1.4.2 Note the poor estimates of the albedo (and surface normal) in the area surrounding the nostrils. What is the source of this error? Describe one method for finding a better estimate of this information from these seven images.**

The main problem is the shadow being omitted in the model which leads to poor result of least square method.

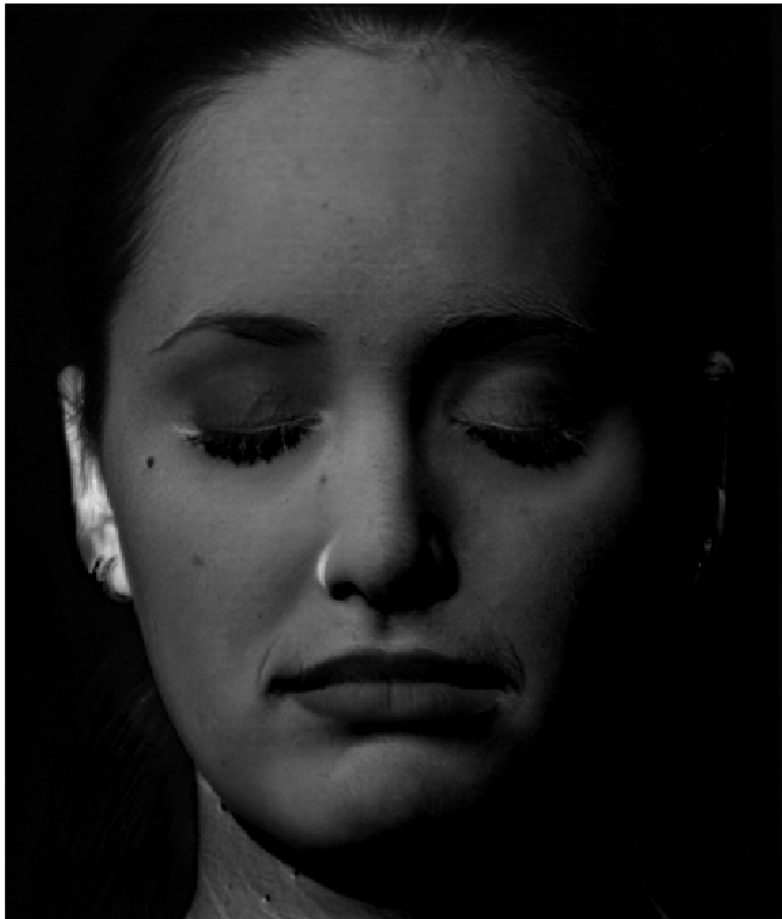
One solution may be masking out part of those images that contain shadow and calculate the albedo of those part with images that does not shadow those part.

**1.4.3 Use the recovered surface information to predict what the person would look like (in grayscale) if illuminated from direction  $\hat{s} = (0.58, -0.58, -0.58)$  and from direction  $\hat{s} = (-0.58, -0.58, -0.58)$ . Submit your results and your code.**

$\vec{s} = (0.58, -0.58, -0.58)$  :

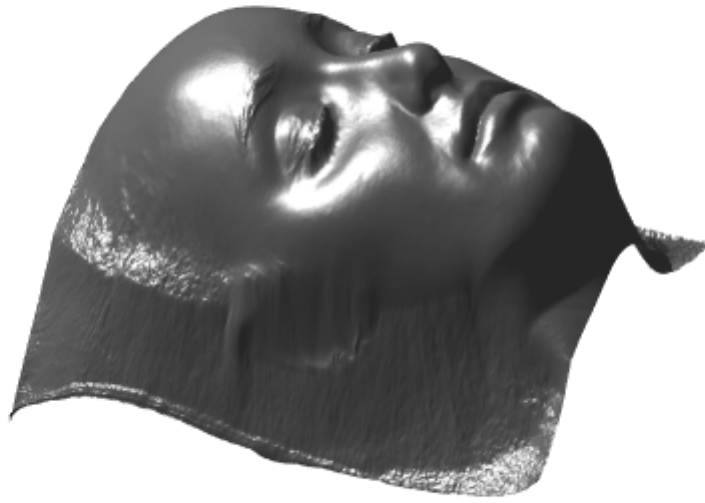


$\vec{s} = (-0.58, -0.58, -0.58) :$



**The function `integrate frankot.m` in the `matlab` folder can be used to recover a surface  $z(x, y)$  from your surface normals  $\hat{n}(x, y)$ , and the surface can be displayed in Matlab. Display and submit two views of the recovered shape.**

code used in this part comes from the hint part of Harvard cs283 assignment 7  
<https://www.seas.harvard.edu/courses/cs283/assgn7.pdf>



## 1.5 Dichromatic reflectance

**1.5.1** Assuming that the spectral sensitivities of a camera's three filters are  $(c_R(\lambda), c_G(\lambda), c_B(\lambda))$  and that the BRDF at the surface point imaged at pixel  $\vec{u}$  is  $f(\lambda, \hat{\omega}_i(\vec{u}), \hat{\omega}_o(\vec{u})) = f_d(\lambda, \vec{u}) + f_s(\hat{\omega}_i(\vec{u}), \hat{\omega}_o(\vec{u}))$ , write expressions for the elements of the diffuse color vectors  $\vec{d}(\vec{u})$  and the source color vector  $f_s$ .

- Diffuse term varies with wavelength, constant with polarization

- Specular term constant with wavelength, varies with polarization

$$\begin{aligned}
\vec{C}(\vec{u}) &= \langle \vec{n}(\vec{u}), \vec{l} \rangle \int_{\lambda} \langle c_R(\vec{u}), c_G(\vec{u}), c_B(\vec{u}) \rangle \circ I(\lambda) (f_d(\lambda) + f_s(\hat{w}_i, \hat{w}_o)) d\lambda \\
&= \langle \vec{n}(\vec{u}), \vec{l} \rangle \int_{\lambda} \langle c_R(\vec{u}), c_G(\vec{u}), c_B(\vec{u}) \rangle \circ I(\lambda) f_d(\lambda) d\lambda + \\
&\quad \langle \vec{n}(\vec{u}), \vec{l} \rangle \int_{\lambda} \langle c_R(\vec{u}), c_G(\vec{u}), c_B(\vec{u}) \rangle \circ I(\lambda) f_s(\hat{w}_i, \hat{w}_o) d\lambda \\
&= \langle \vec{n}(\vec{u}), \vec{l} \rangle \int_{\lambda} \langle c_R(\vec{u}), c_G(\vec{u}), c_B(\vec{u}) \rangle \circ I(\lambda) f_d(\lambda) d\lambda + \\
&\quad f_s(\hat{w}_i, \hat{w}_o) \langle \vec{n}(\vec{u}), \vec{l} \rangle \int_{\lambda} \langle c_R(\vec{u}), c_G(\vec{u}), c_B(\vec{u}) \rangle \circ I(\lambda) d\lambda
\end{aligned}$$

where  $\circ$  is the Hadamard product (element wise product)

Then  $\vec{d}(\vec{u}) = \int_{\lambda} \langle c_R(\vec{u}), c_G(\vec{u}), c_B(\vec{u}) \rangle \circ I(\lambda) f_d(\lambda) d\lambda$  and  
 $\vec{s} = \int_{\lambda} \langle c_R(\vec{u}), c_G(\vec{u}), c_B(\vec{u}) \rangle \circ I(\lambda) d\lambda$

**1.5.2 Suppose you are given two unit-length three-vectors  $\hat{r}_1$  and  $\hat{r}_2$  that are orthogonal to  $\vec{s}$ . Show that the two-channel image<sup>1</sup> given by the per-pixel inner products ( $\langle \hat{r}_1, \hat{C}(\hat{u}) \rangle, \langle \hat{r}_2, \hat{C}(\hat{u}) \rangle$ ):**

**a ) does not depend on the specular components of the BRDFs,  $f_s(\hat{w}_i(\vec{u}), \hat{w}_o(\vec{u}))$**

**b) depends linearly on the surface normals,  $\hat{n}(\vec{u})$ .**

$$\begin{aligned}
&\langle \hat{r}_1, \hat{C}(\vec{u}) \rangle \\
&= \langle \hat{r}_1, \langle \vec{n}(\vec{u}), \vec{l} \rangle \vec{d}(\vec{u}) + g_s(\vec{u}) \vec{s} \rangle \\
&= \langle \hat{r}_1, \langle \vec{n}(\vec{u}), \vec{l} \rangle \vec{d}(\vec{u}) \rangle + \langle \hat{r}_1, \langle \vec{n}(\vec{u}), g_s(\vec{u}) \rangle \vec{s} \rangle
\end{aligned}$$

where  $\vec{r}_1 \perp \vec{s}$  and  $\vec{r}_2 \perp \vec{s}$ , then the channel is independent to  $g_s(\vec{u})$  which depends on the specular component of BRDFs

$$\begin{aligned}
&\text{since } \langle \hat{r}_1, \hat{C}(\vec{u}) \rangle \\
&= \langle \hat{r}_1, \langle \vec{n}(\vec{u}), \vec{l} \rangle \vec{d}(\vec{u}) \rangle + 0 \\
&= \langle \vec{n}(\vec{u}), \vec{l} \rangle \langle \hat{r}_1, \vec{d}(\vec{u}) \rangle
\end{aligned}$$

Then since  $\vec{d}(\vec{u})$  is independent of  $\vec{n}(\vec{u})$ , the two channels depends linearly on the inner product of surface normal assumes that  $l(\vec{u})$  is fixed,  $\hat{n}(\vec{u})$

**1.5.3 Show that the two properties from part (b) are also satisfied by the single-channel (gray scale) image  $J(\vec{u}) = ||J(\vec{u})||$ .**

let  $\langle \vec{n}(\vec{u}), \vec{l} \rangle = k$ , then

$$\begin{aligned}
J(\vec{u}) &= ||\vec{J}(\vec{u})|| \\
&= \sqrt{(k \langle \hat{r}_1, \vec{d}(\vec{u}) \rangle)^2 + (k \langle \hat{r}_2, \vec{d}(\vec{u}) \rangle)^2}
\end{aligned}$$

$$= \sqrt{k^2 ((\langle \hat{r}_1, \vec{d}(\vec{u}) \rangle)^2 + (\langle \hat{r}_1, \vec{d}(\vec{u}) \rangle)^2)}$$

$$= k \sqrt{(\langle \hat{r}_1, \vec{d}(\vec{u}) \rangle)^2 + (\langle \hat{r}_1, \vec{d}(\vec{u}) \rangle)^2}$$

is also linear dependent to  $\langle \vec{n}(\vec{u}, \vec{l}) \rangle$ , while since they both are independent of  $g_s$ , their composition must also be independent of  $g_s L_f \vec{l}$

**1.5.4 Write a function `imout=makeLambertian(im,s)` that takes an RGB image `im` and a source color vector `s` and computes the grayscale image `J(~u)` from part (c). Run this function on the image `fruitbowl.png` from the data folder using `s=[0.6257 0.5678 0.5349]` and submit the results. Explain in three sentences or less why `imout` might be more useful than `im` to a computer vision system. Note that a simple way to numerically obtain a set of  $N - 1$  unit-length vectors that are orthogonal to  $N$ -vector is to use the null command in Matlab.**

why `imout` might be more useful than `im`: Specular depends on light source and view point while diffusion depends mostly on the object itself which make visual recognition and tracking easier. Moreover, there will be less sharp changes in gradient due to light(direct reflect of specular term) which is beneficial for most visual system.



## 1.6 Color metamers (Extra Credit)

**1.6.1 Show that the tristimulus vector  $\vec{C}_f = (X_f, Y_f, Z_f)$  for a ripe banana  $f(\lambda)$  under illuminant spectrum  $l(\lambda)$  can be written as  $\vec{C}_f = L_f \vec{l}$  where  $L_f$  is a  $3 \times N$  matrix and  $\vec{l}$  is the discretization of  $l(\lambda)$ .**

$$\begin{aligned}\vec{C}_f &= (X_f, Y_f, Z_f) = \int_{\lambda} \vec{C}(\lambda) f(\lambda) l(\lambda) d\lambda \\ &= R \circ \begin{bmatrix} | & & | \\ f(\lambda_1) & \dots & f(\lambda_N) \\ | & & | \end{bmatrix} \vec{l} = L_f \vec{l}\end{aligned}$$

**1.6.2 Given  $\vec{f}$  and  $\vec{g}$ , a good way to choose  $\vec{l}$  is to minimize the distance, in a least-squares sense, of the two resulting tristimulus vectors. Write an expression for the Euclidean distance between the two tristimulus vectors in terms of  $L_f$ , its counterpart  $L_g$ , and  $\vec{l}$ .**

$$\|L_f \vec{l} - L_g \vec{l}\|$$

**1.6.4 The file bananas.mat from the data folder contains two spectral reflectance functions, ripe and overripe, which are sampled in 10nm increments from 400nm to 700nm and correspond roughly to the data from Day 5 and Day 7 of Figure 4. Use your function to find the optimal metamer-inducing blackbody temperature for these materials, and plot the distance as a function of temperature, and the spectral power distribution of the best illuminant. You will need to use the file cixyz64 1.csv from the data folder, which contains tabulated data for the CIE XYZ color matching functions. In Matlab, you can read data from a CSV file using the csvread command. Remember to normalize the spectral power distributions of blackbody radiators in a way that is meaningful for the minimization criterion of part (b).**

best  $T = 8300K$  with precision  $\pm 50K$



