

计算机视觉：三维重建

郭亨

计算机视觉中的三维重建

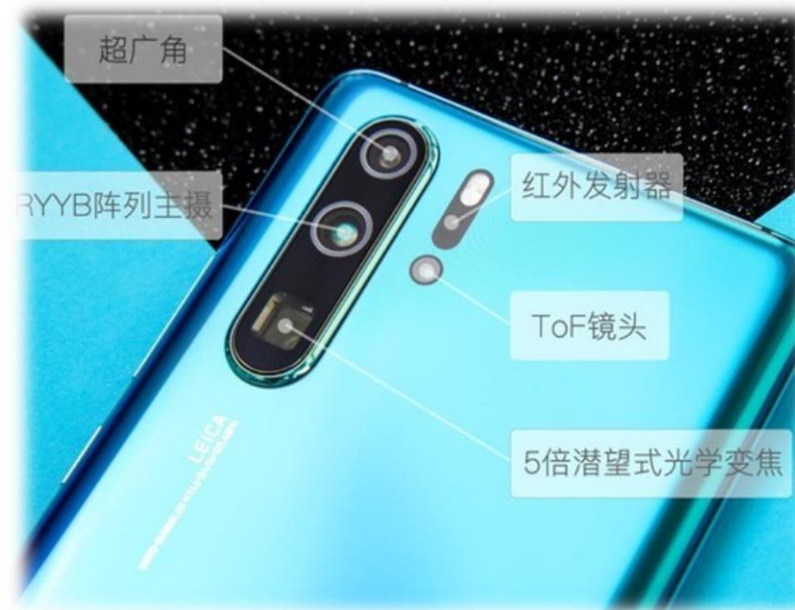
三维重建：通过相机拍摄图像获取三维形状，计算机视觉的研究热点



苹果 Iphone



三星 Galaxy Note 10



华为 P30 pro

基于几何方法的三维重建



多视角立体视觉
Multiview stereo



重建结果
[Furukawa 10]



真值



光度方法三维重建

(输入多光照图像、输出三维)

3D Scanning the President of the United States
P. Debevec et al., USC, 2014



几何（多视角） vs. 光度（多光照）

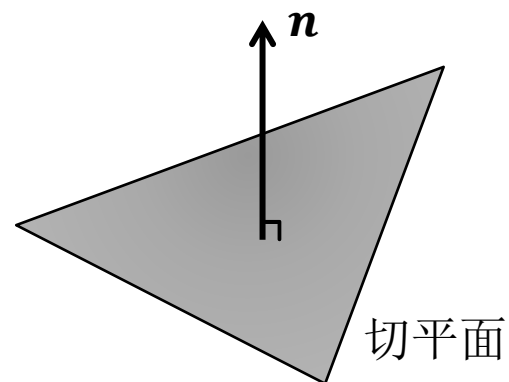
	几何方法 Geometric	光度方法 Photometric
完整形状	○	×
形状细节	×	○
		

预备知识：表面法线

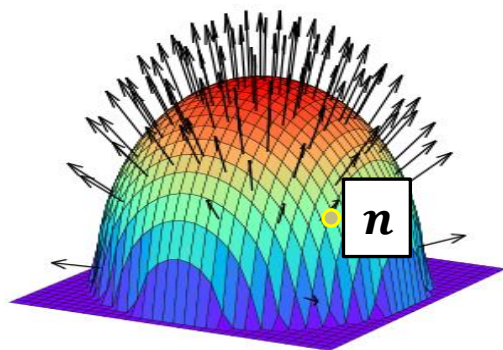
曲面某个点的法线 \mathbf{n} 是垂直于该点切平面的单位向量


$$\mathbf{n} \in S^2 \subset \mathbb{R}^3, \|\mathbf{n}\| = 1$$

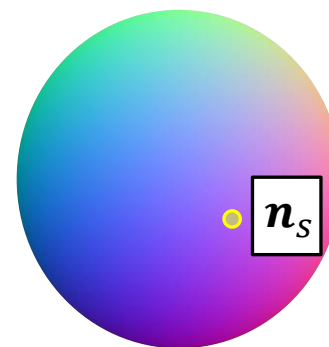
$$\mathbf{n} = \begin{bmatrix} n_x \\ n_y \\ n_z \end{bmatrix}$$



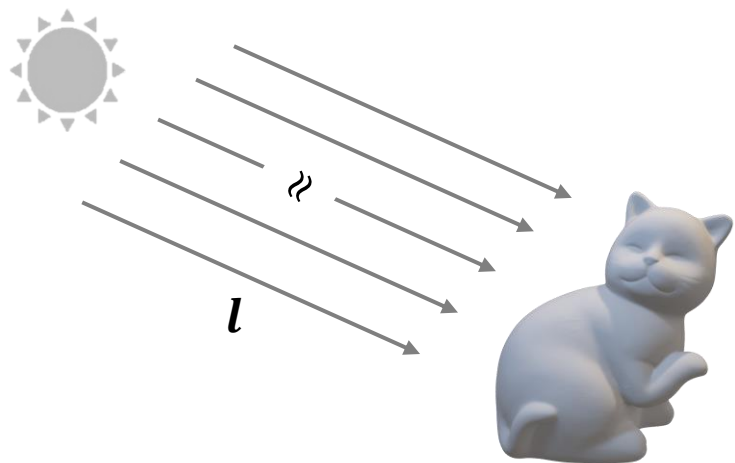
曲面的法线 \mathbf{n} 通常用伪彩色图 \mathbf{n}_s 来可视化



$$\mathbf{n}_s = \frac{\mathbf{n} + 1}{2}$$




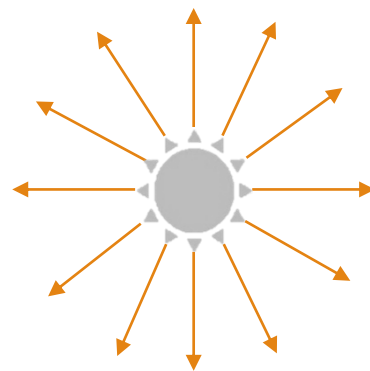
预备知识：平行光源



所有点接收到的光源方向一致

$$l \in S^2 \subset \mathbb{R}^3, \|l\| = 1$$

$$l = \begin{bmatrix} l_x \\ l_y \\ l_z \end{bmatrix}$$



假设没有环境光（暗室）

点光源为全方向光源

无穷远处点光源可以近似为平行光

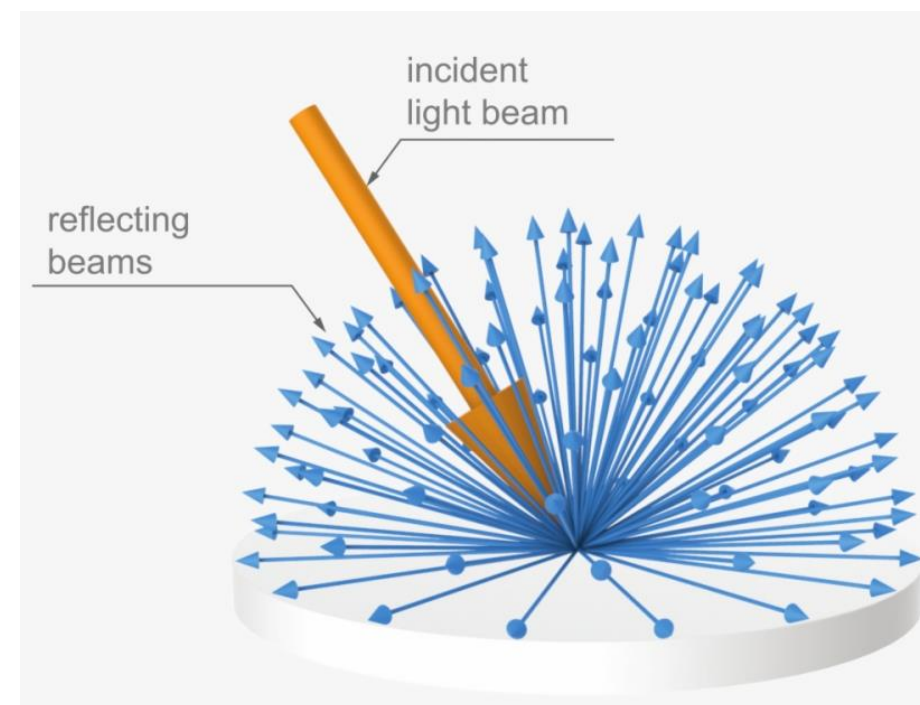
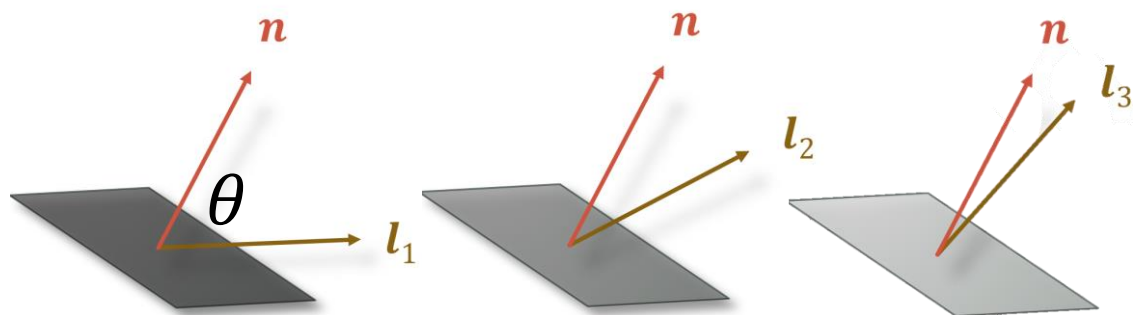


Johann Heinrich Lambert (1728–1777)

约翰·海因里希·朗伯

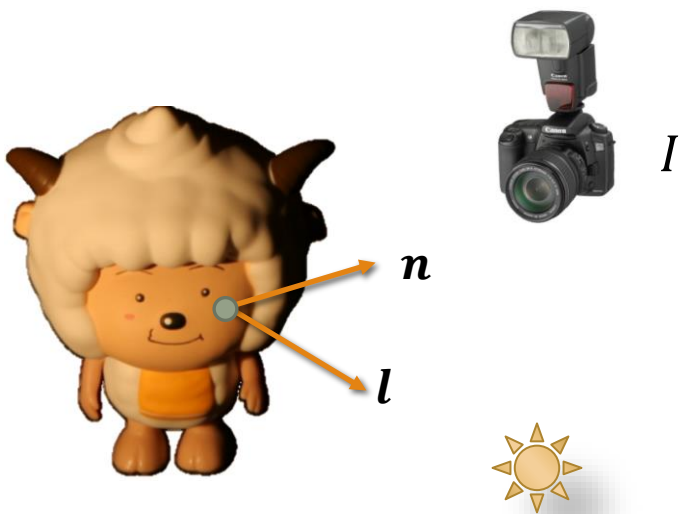
预备知识：朗伯反射

- 理想的漫反射
- 反射光强与视角无关
- 朗伯表面反射的光强正比于 $\mathbf{n}^\top \mathbf{l} (= \cos \theta)$



朗伯成像模型

$$I \propto e\rho \underset{\text{(W/m}^2\text{)}}{\mathbf{n}^\top \mathbf{l}} = e\rho \begin{bmatrix} n_x & n_y & n_z \end{bmatrix} \begin{bmatrix} l_x \\ l_y \\ l_z \end{bmatrix}$$



$I \in \mathbb{R}_+$: 观测到的像素值
 $e \in \mathbb{R}_+$: 光源辐射强度
 $\rho \in \mathbb{R}_+$: 朗伯反射的反射率
 \mathbf{n} : 表面法线方向 (3维单位向量)
 \mathbf{l} : 光源方向 (3维单位向量)

朗伯成像模型

$$I \propto e\rho \mathbf{n}^T \mathbf{l} = e\rho [n_x \quad n_y \quad n_z] \begin{bmatrix} l_x \\ l_y \\ l_z \end{bmatrix}$$

假设光源强度 e 为1
暂且忽略 ρ 的影响



$$I = \mathbf{n}^T \mathbf{l} = \mathbf{n} \cdot \mathbf{l}$$

光度立体视觉/Photometric stereo



输入

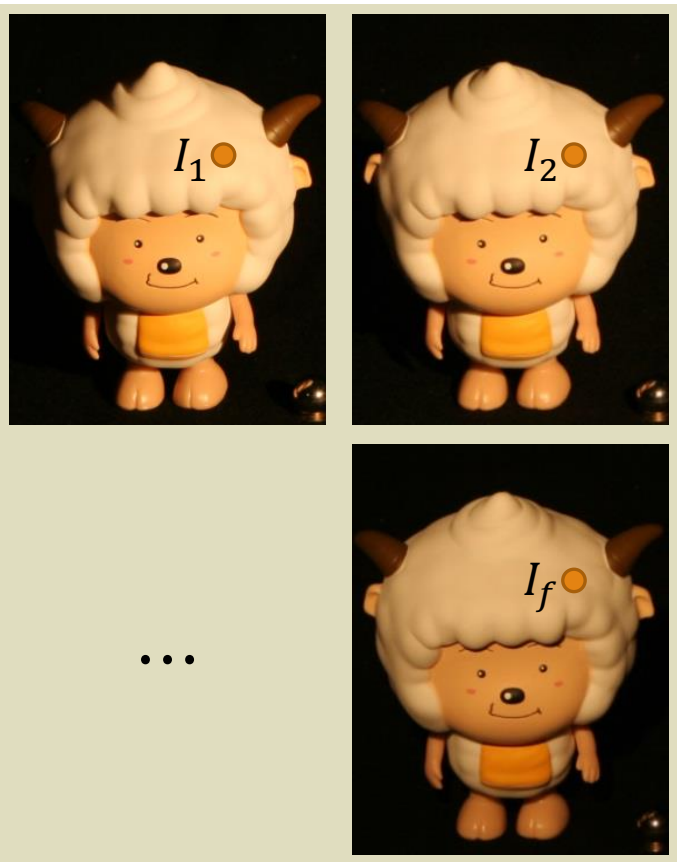
假设物体表面反射符合朗伯成像模型，给定多张不同平行光光照下的图片 (I) 及其光源方向 (L)，求解每个像素的法线方向 (N)。



输出

[Woodham, 80]

光度立体视觉/Photometric stereo



考虑单个空间点
其表面法线为 \mathbf{n}

$$\begin{aligned} I_1 &= \mathbf{l}_1^\top \mathbf{n} \\ I_2 &= \mathbf{l}_2^\top \mathbf{n} \\ &\vdots \\ I_f &= \mathbf{l}_f^\top \mathbf{n} \end{aligned}$$

$$\begin{bmatrix} I_1 \\ I_2 \\ \vdots \\ I_f \end{bmatrix} = \begin{bmatrix} l_{1x} & l_{1y} & l_{1z} \\ l_{2x} & l_{2y} & l_{2z} \\ \vdots & \vdots & \vdots \\ l_{fx} & l_{fy} & l_{fz} \end{bmatrix} \begin{bmatrix} n_x \\ n_y \\ n_z \end{bmatrix}$$

$$\mathbf{L} \in \mathbb{R}^{f \times 3}$$

一共有 f 张图像（ f 种不同光源变化）

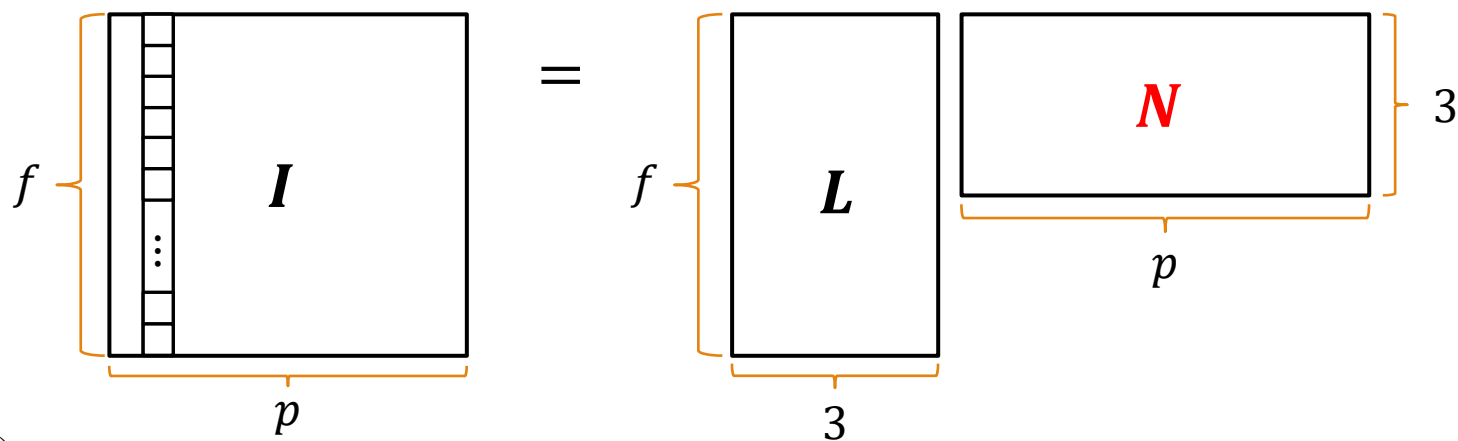
光度立体视觉/Photometric stereo



一共有 f 张图像（ f 种不同光源变化）

考虑整体图像
 $I = LN$

p : 像素个数
 f : 图像张数



光度立体视觉/Photometric stereo



$$\begin{matrix} f \\ \left\{ \right. \end{matrix} \begin{matrix} \square \\ \mathbf{I} \\ \square \end{matrix} \begin{matrix} \left. \right\} \\ p \end{matrix} = \begin{matrix} f \\ \left\{ \right. \end{matrix} \begin{matrix} \square \\ \mathbf{L} \\ \square \end{matrix} \begin{matrix} \left. \right\} \\ 3 \end{matrix} \begin{matrix} \square \\ \mathbf{N} \\ \square \end{matrix} \begin{matrix} \left. \right\} \\ p \end{matrix} \begin{matrix} \left. \right\} \\ 3 \end{matrix}$$

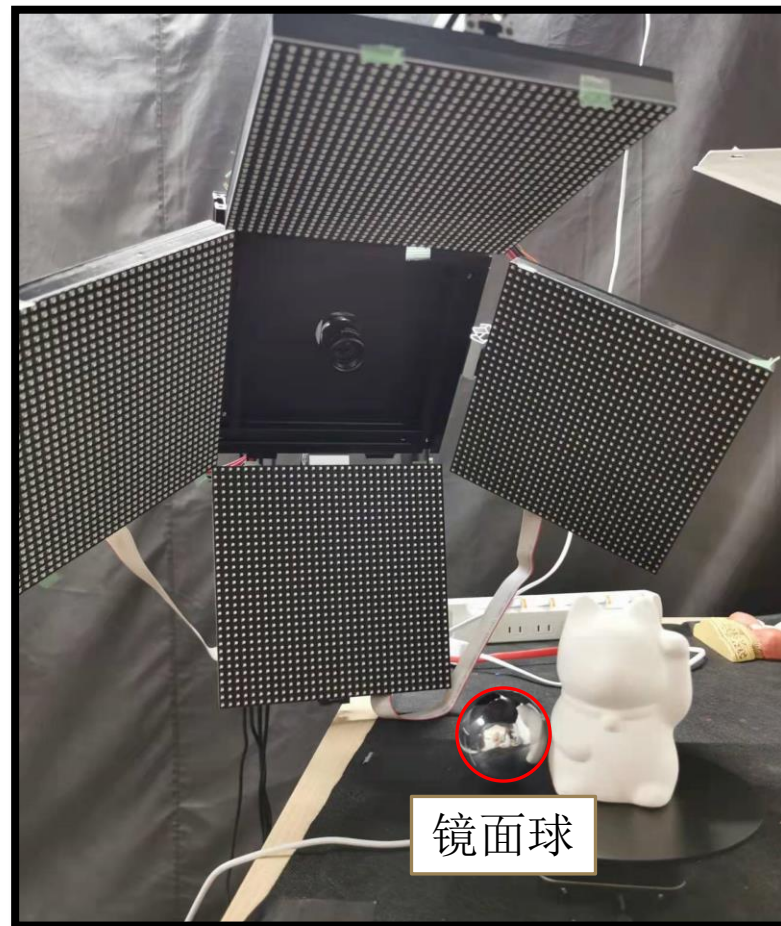
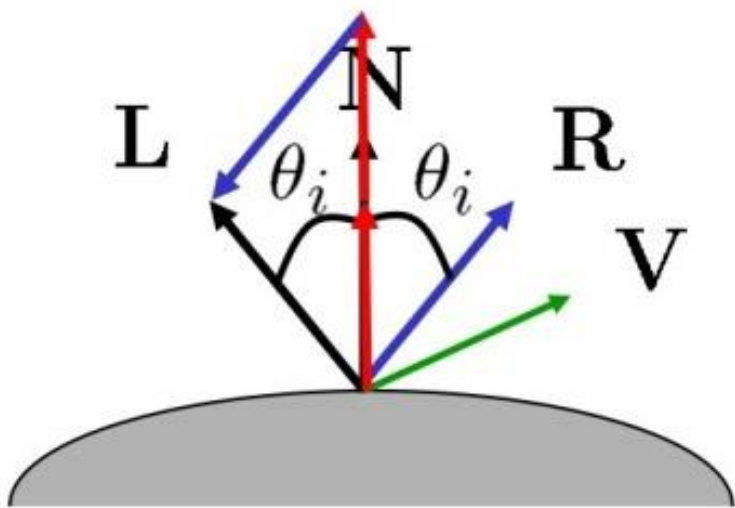
最小二乘法求解:

$$\mathbf{N} = \mathbf{L}^\dagger \mathbf{I}$$

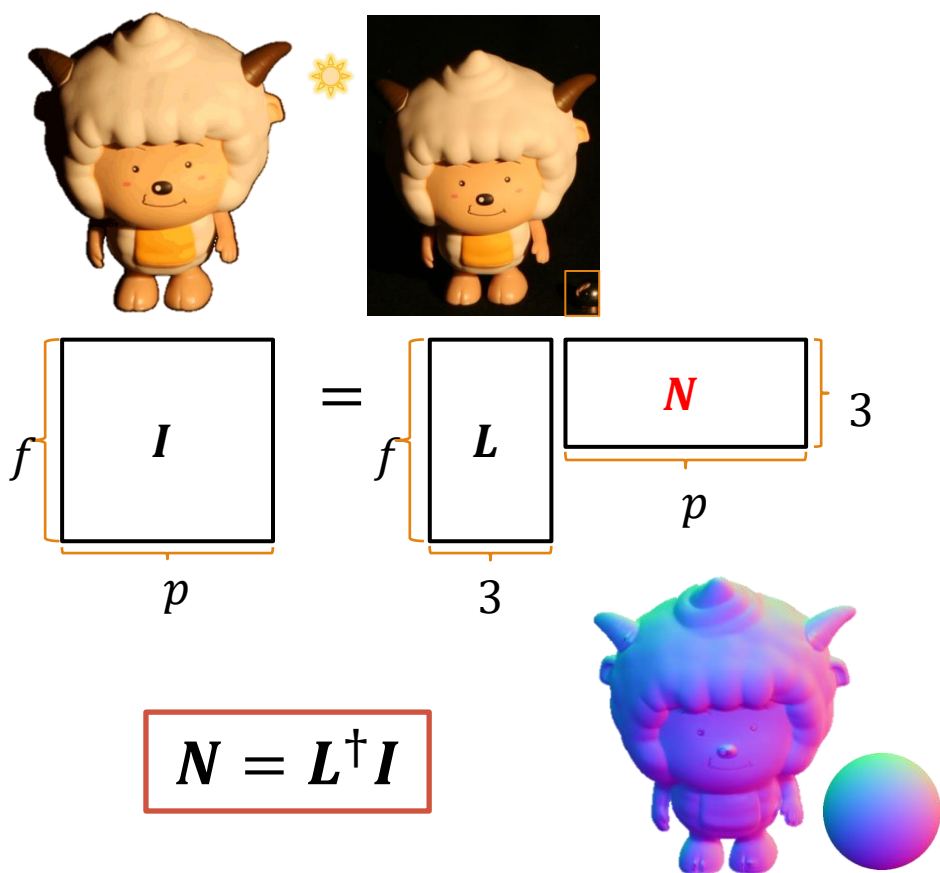
一共有 f 张图像（ f 种不同光源变化）

如何获取平行光源的方向

用镜面球进行光源标定



光度立体完整流程



1. 读图像

2. 标光源

3. 求法线

4. 看结果

```
import numpy as np
import cv2
import glob
from sklearn.preprocessing import normalize
```

```
# Read images
```

```
M = []
```

```
for fname in sorted(glob.glob('data/cat/*.png')):
```

```
    im = cv2.imread(fname, 0)
```

```
    if M == []:
```

```
        height, width = im.shape
```

```
        M = im.reshape((-1, 1))
```

```
    else:
```

```
        M = np.append(M, im.reshape((-1, 1)), axis=1)
```

```
# Read light directions
```

```
Lt = np.loadtxt('data/cat/lights.txt') # Transpose of L
```

```
# Photometric stereo computation (least-squares)
```

```
### Please fill the code here
```

```
# Visualization
```

```
N = np.reshape(N, (height, width, 3)) # Reshape to image coordinates
```

```
N[:, :, 0], N[:, :, 2] = N[:, :, 2], N[:, :, 0].copy() # Swap RGB <-> BGR
```

```
N = (N + 1.0) / 2.0 # Rescale
```

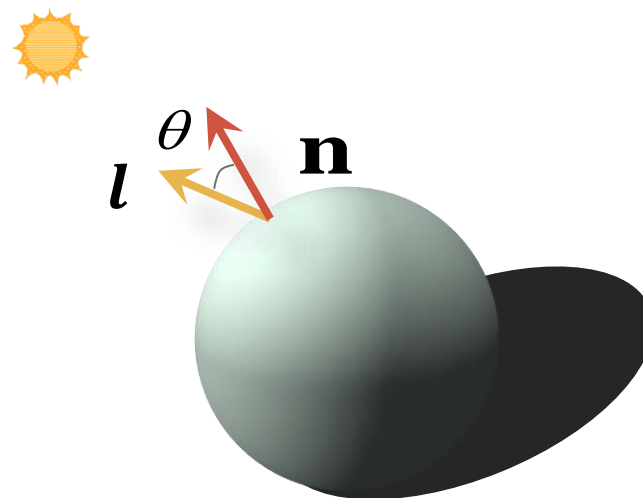
```
cv2.imshow('normal map', N)
```

```
cv2.waitKey()
```

```
cv2.destroyAllWindows()
```

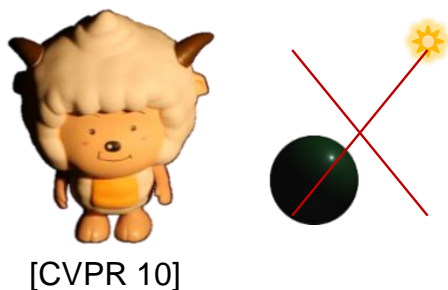
小结

- 一个对比
 - 几何方法 vs 光度方法
- 三个概念
 - 表面法线 \mathbf{n} 、平行光源 \mathbf{l} 、朗伯反射 $\mathbf{n}^\top \mathbf{l}$
- 四个步骤
 - 读图像、标光源、求法线、看结果

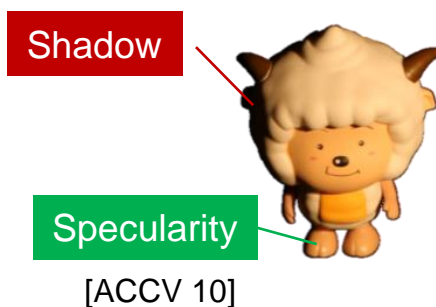


思考: 经典光度立体方法中的问题

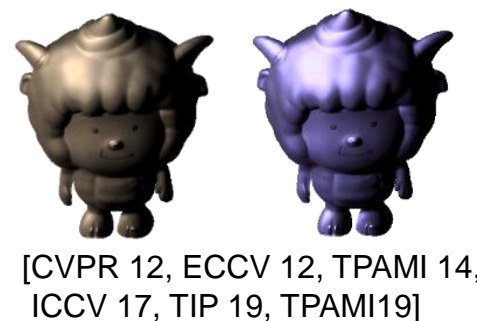
方向1: 非标定光源



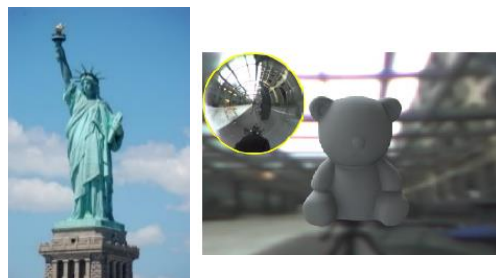
方向2: 鲁棒求解



方向3: 复杂反射与真实材质



方向4: 自然光照



方向5: 非标定光源 + 真实材质

