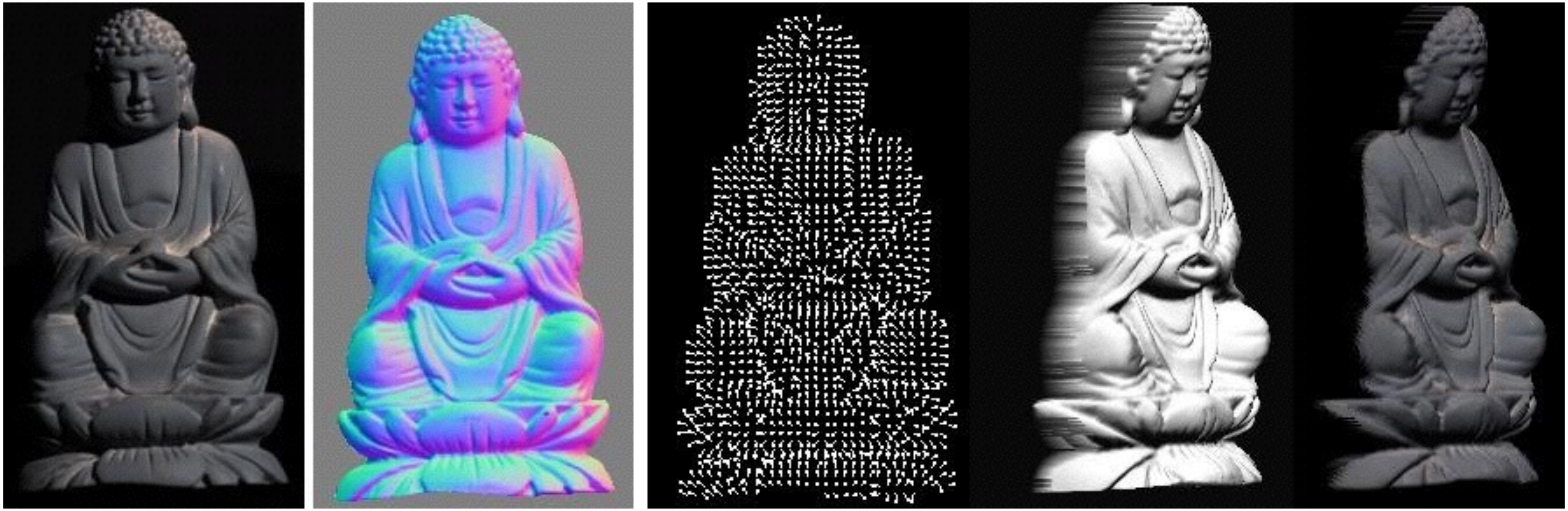
# CS4670/5760: Computer Vision

~~Kavita Bala~~ Scott Wehrwein

## Lecture 23: Photometric Stereo



# Announcements

- PA3 Artifact due tonight
- PA3 Demos Thursday
  - Signups close at 4:30 today
- No lecture on Friday

# Last Time: Two-View Stereo

# Last Time: Two-View Stereo

Key Idea: use feature motion to understand shape

# Today: Photometric Stereo

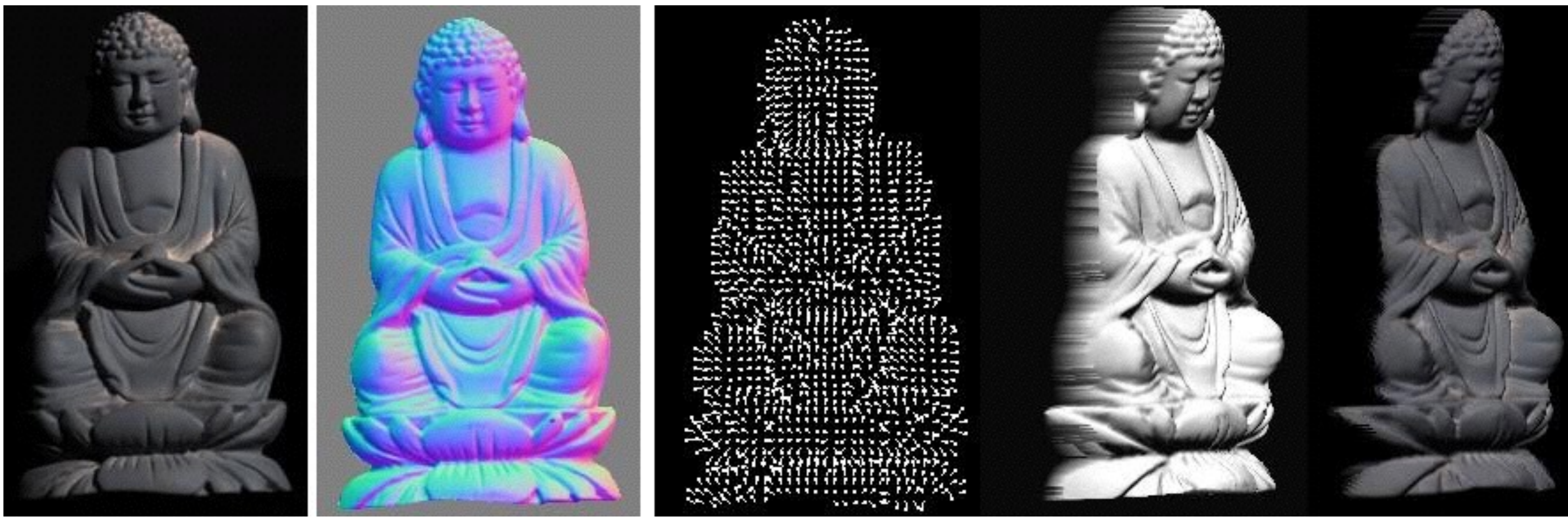
Key Idea: use pixel brightness to understand shape

# Today: Photometric Stereo

Key Idea: use pixel brightness to understand shape

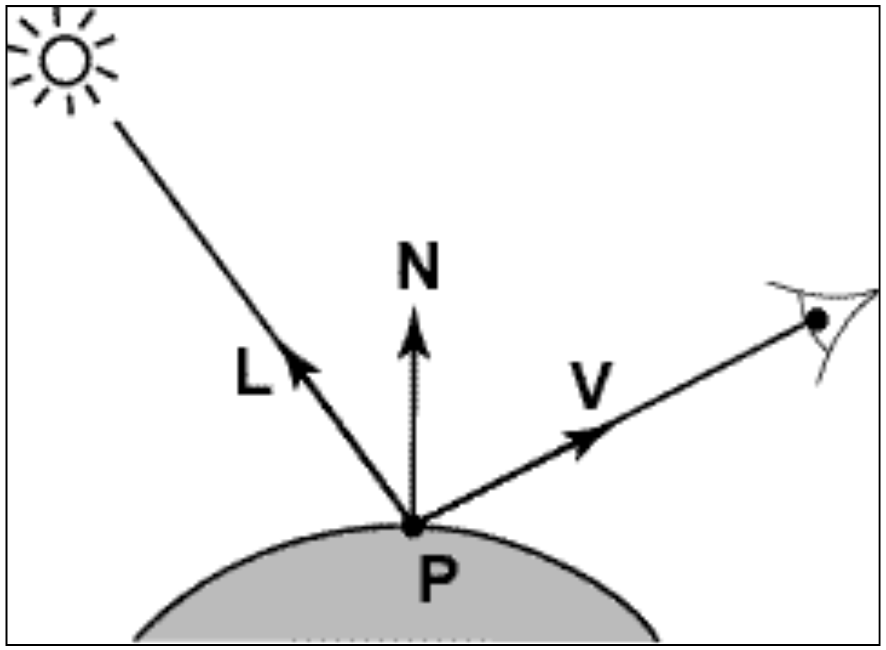
# Photometric Stereo

What results can you get?



Input (1 of 12) | Normals (RGB colormap) | Normals (vectors) | Shaded 3D rendering | Textured 3D rendering

# Modeling Image Formation

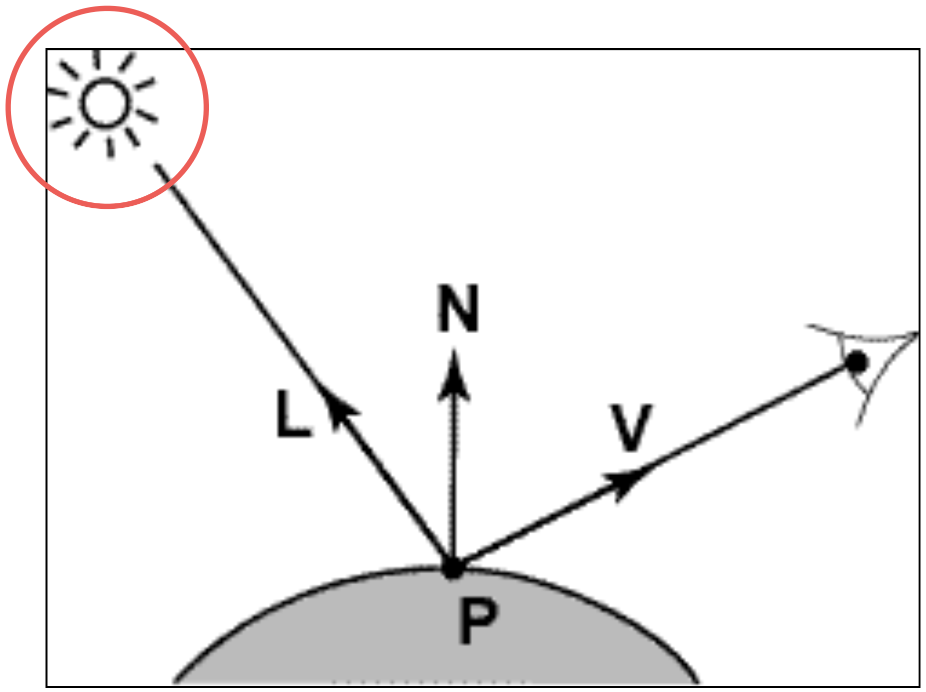



Now we need to reason about:

- How light interacts with the scene
- How a pixel value is related to light energy in the world

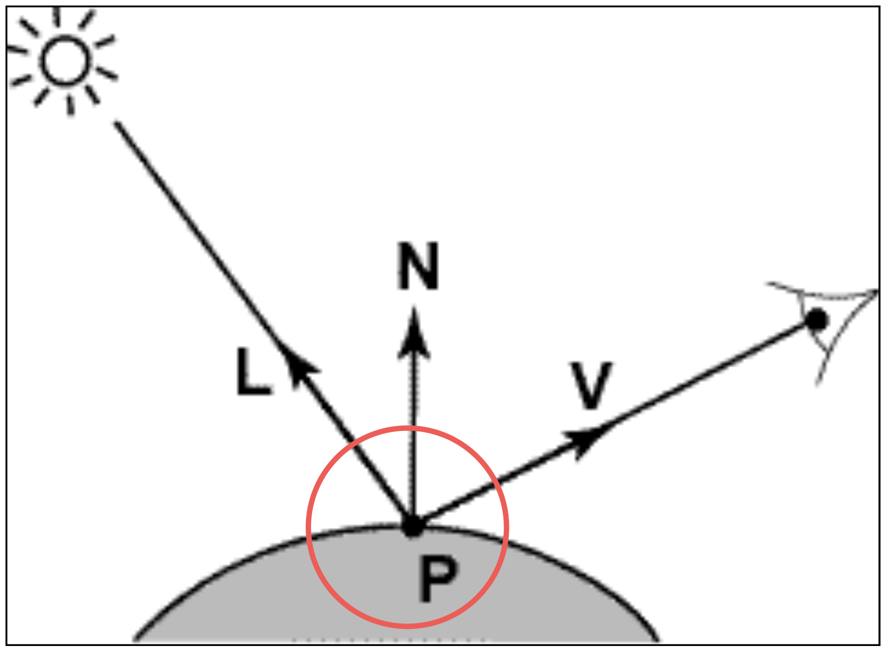
Let's track a "ray" of light all the way from light source to the sensor.

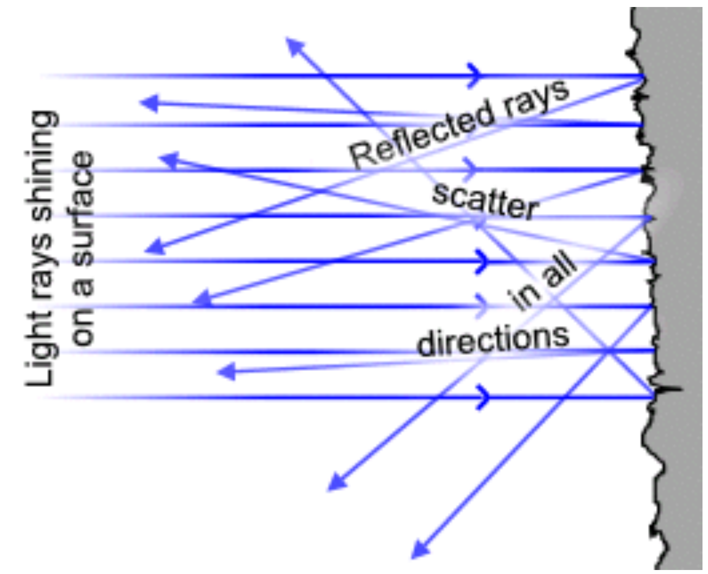
# Directional Lighting




- Key property: all rays are parallel
- Equivalent to an infinitely distant point source

# Lambertian Reflectance



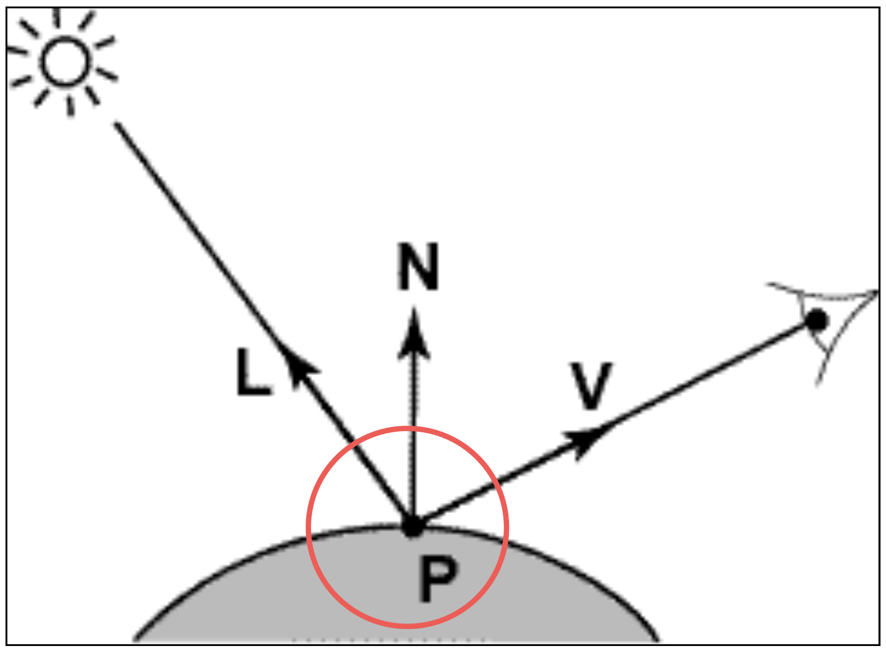



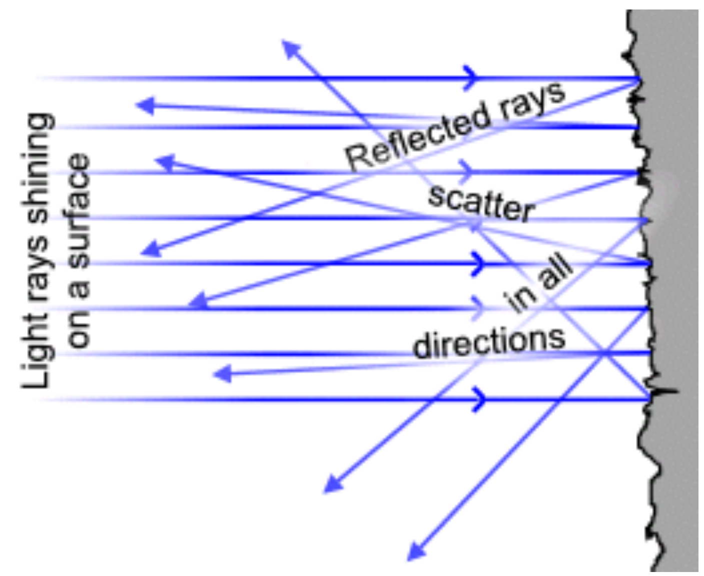

$$I = N \cdot L$$

Image intensity = Surface normal • Light direction

Image intensity $\propto$ cos(angle between N and L)

# Lambertian Reflectance



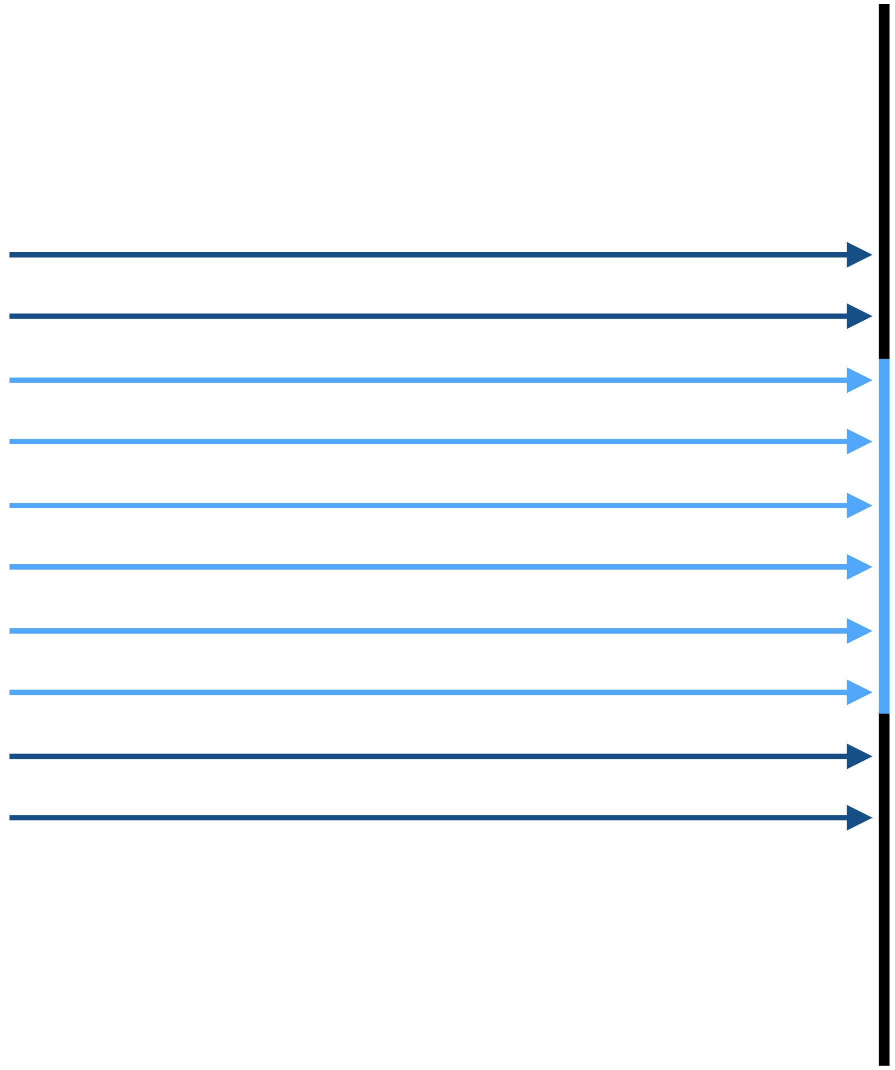




1. Reflected energy is proportional to cosine of angle between L and N **(incoming)**

2. Measured intensity is viewpoint-independent **(outgoing)**

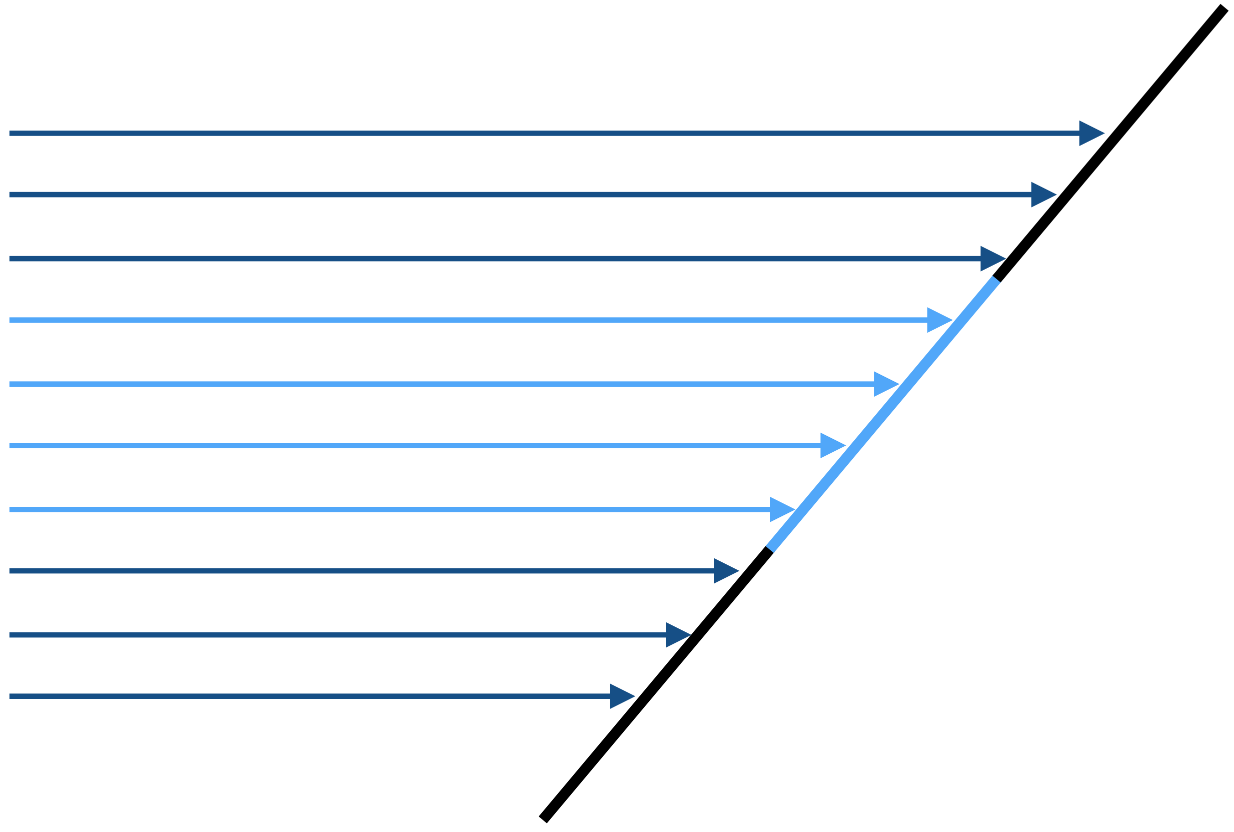
# Lambertian Reflectance: Incoming

1. Reflected energy is proportional to cosine of angle between L and N



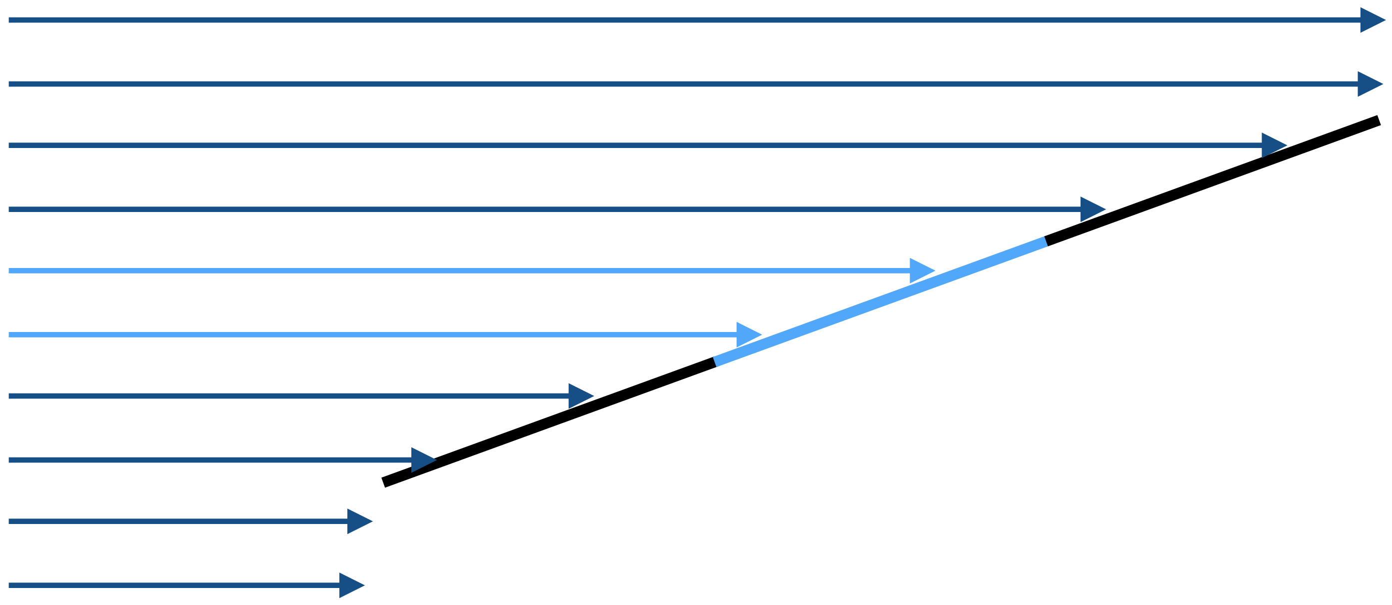
# Lambertian Reflectance: Incoming

1. Reflected energy is proportional to cosine of angle between L and N



# Lambertian Reflectance: Incoming

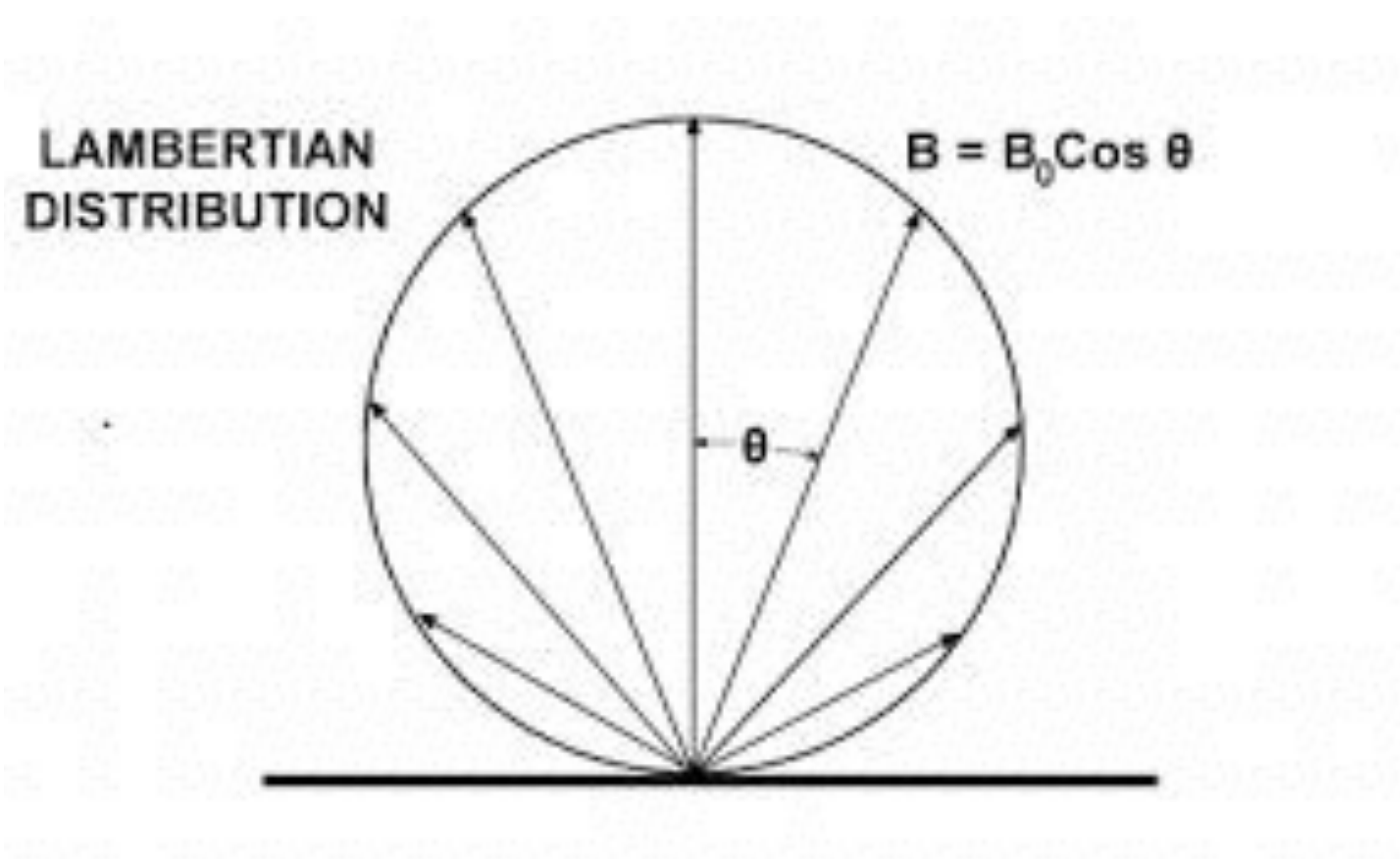
1. Reflected energy is proportional to cosine of angle between L and N



Light hitting surface is proportional to the cosine

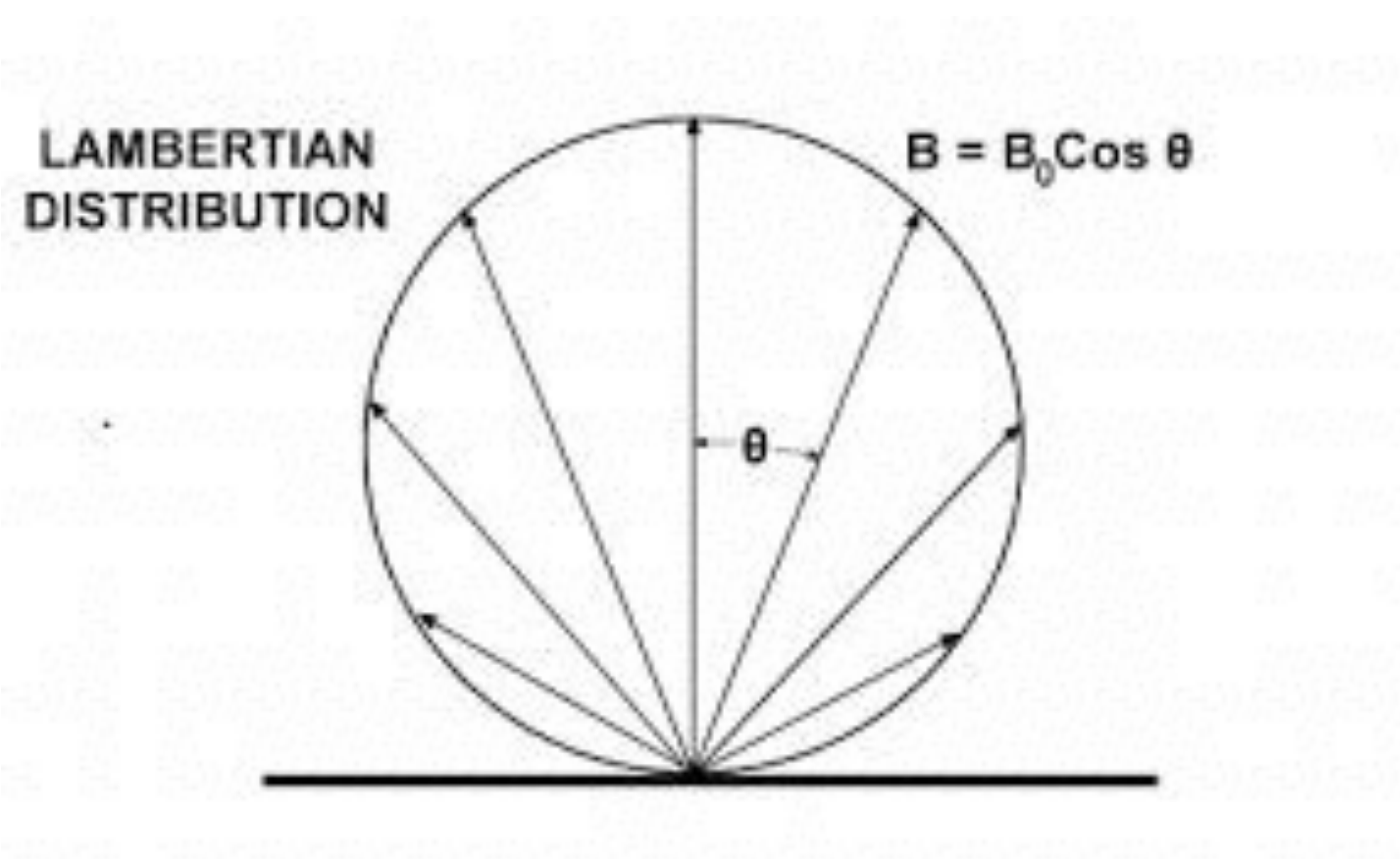
# Lambertian Reflectance: Outgoing

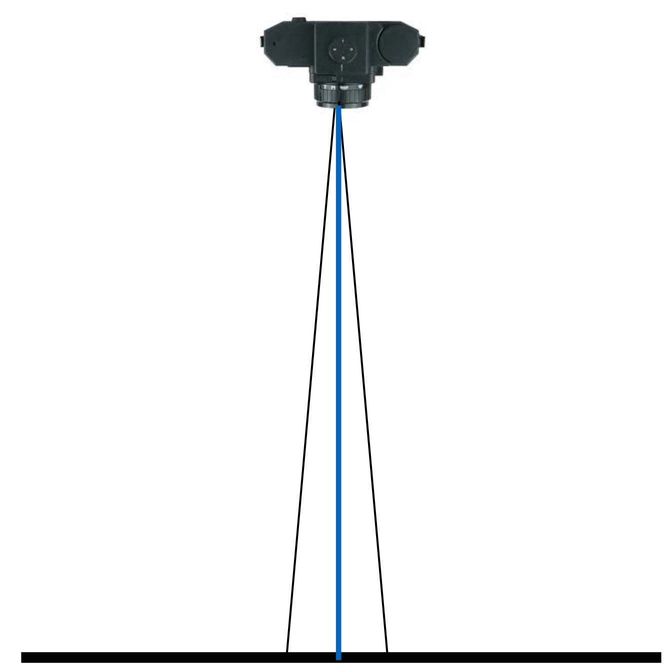
2. Measured intensity is viewpoint-independent



# Lambertian Reflectance: Outgoing

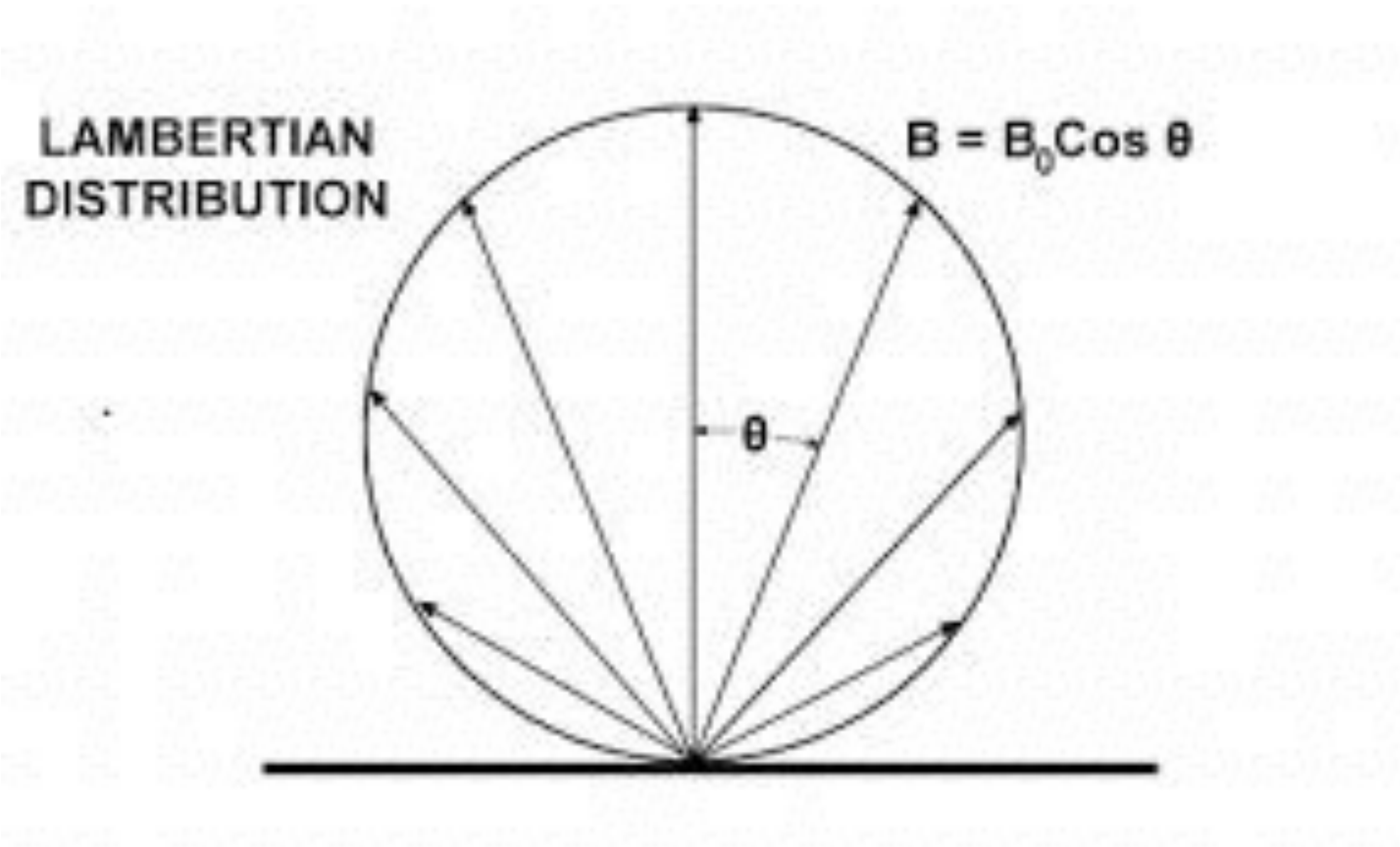
2. Measured intensity is viewpoint-independent

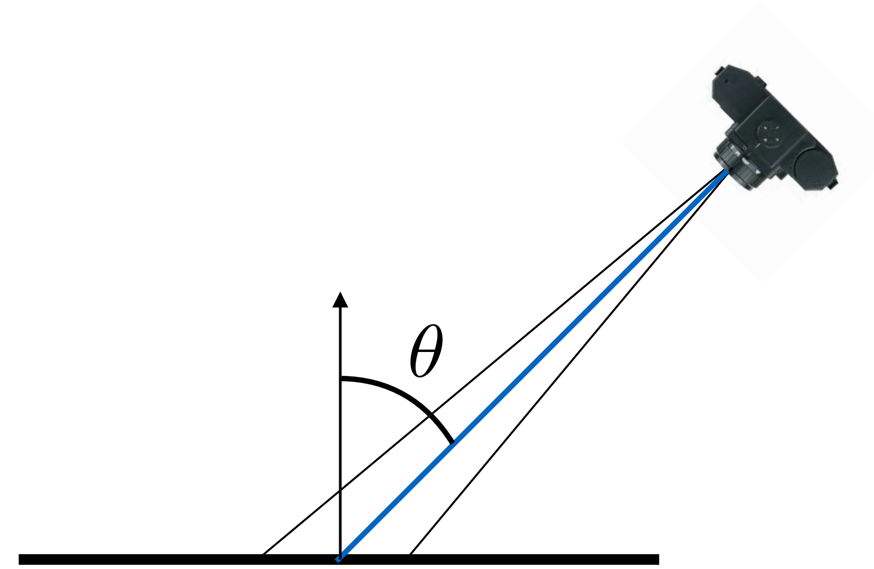


# Lambertian Reflectance: Outgoing

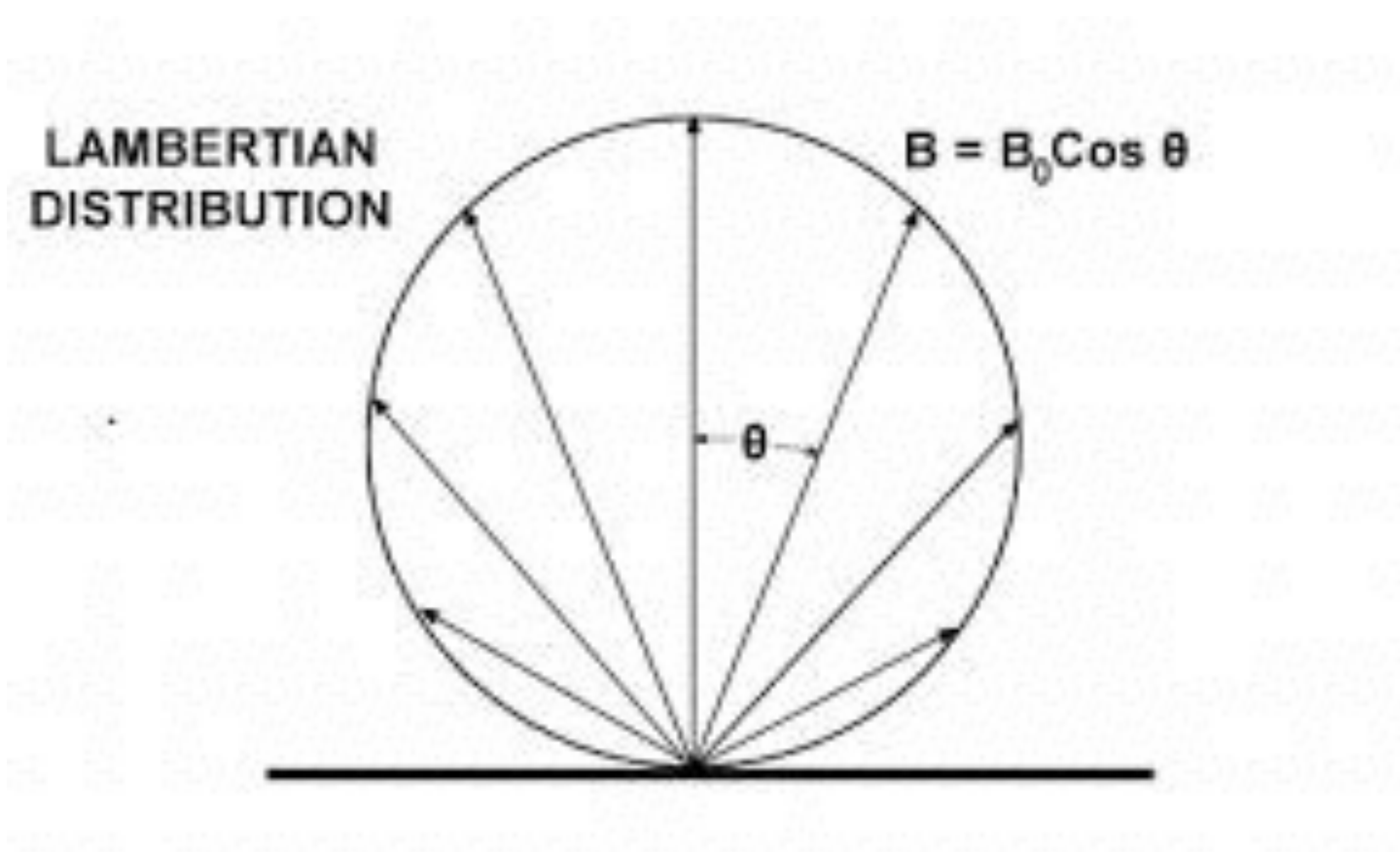
2. Measured intensity is viewpoint-independent

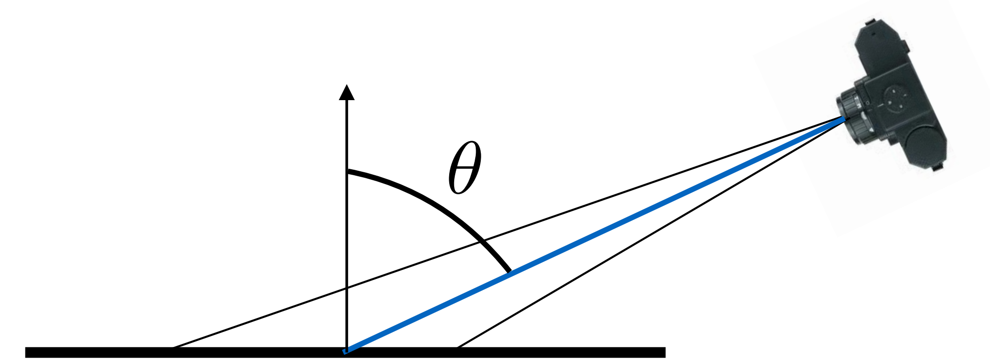


# Lambertian Reflectance: Outgoing

2. Measured intensity is viewpoint-independent

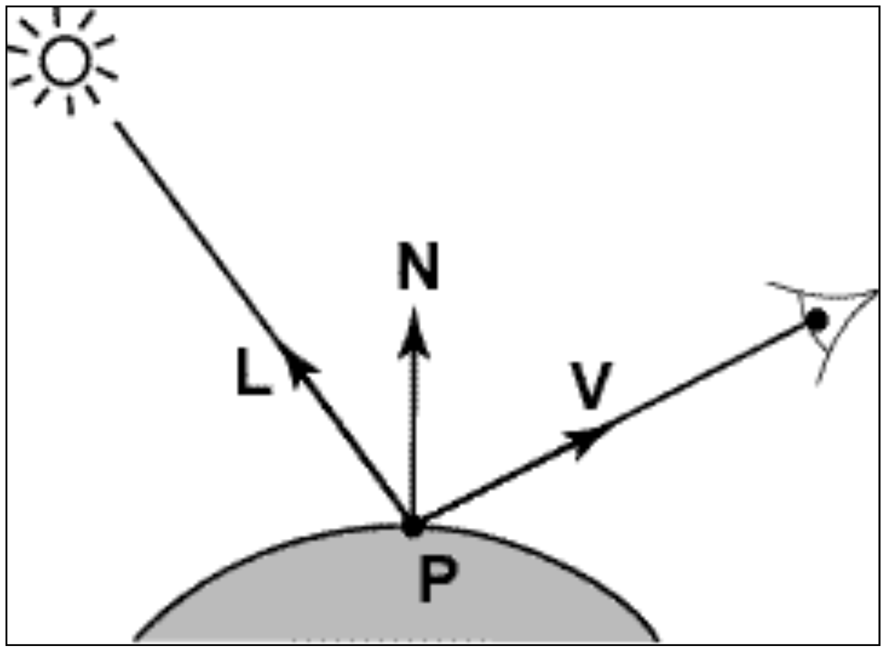



$$A \propto \frac{1}{\cos(\theta)}.$$

Measured intensity $\propto B_0 \cancel{\cos(\theta) \frac{1}{\cos(\theta)}}$

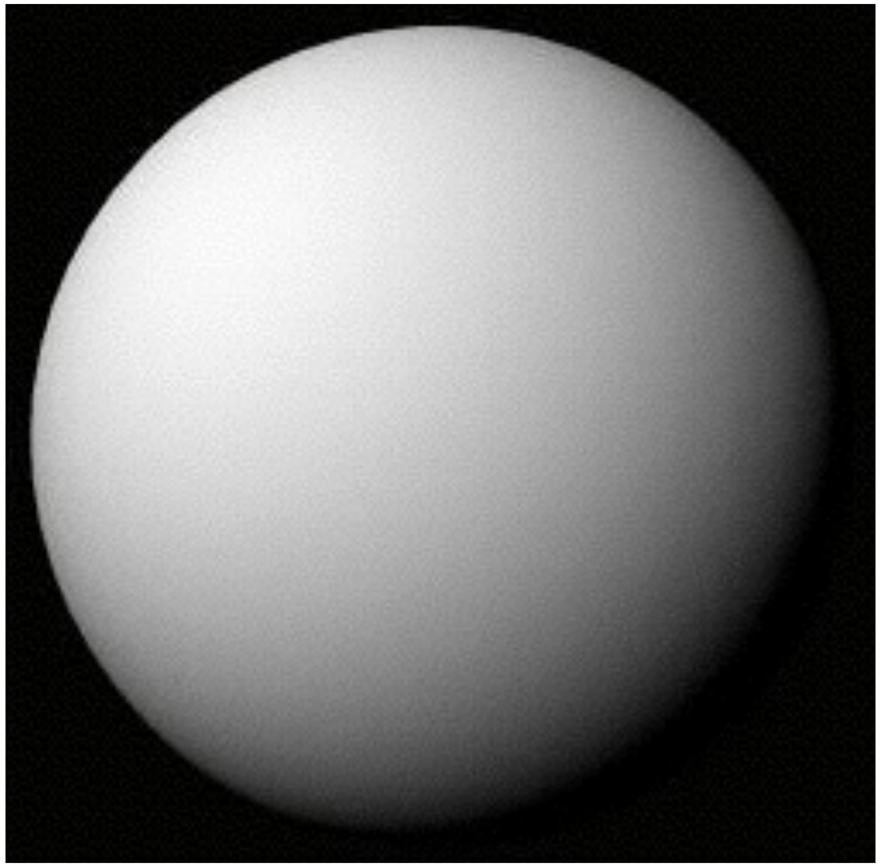
# Image Formation Model: Final




$$I = k_d \mathbf{N} \cdot \mathbf{L}$$

1. Diffuse albedo: what fraction of incoming light is reflected?
    - Introduce scale factor $k_d$
2. Light intensity: how much light is arriving?
    - Compensate with camera exposure (global scale factor)
3. Camera response function
    - Assume pixel value is linearly proportional to incoming energy (perform radiometric calibration if not)

# A Single Image: Shape from Shading



$$I = k_d \mathbf{N} \cdot \mathbf{L}$$

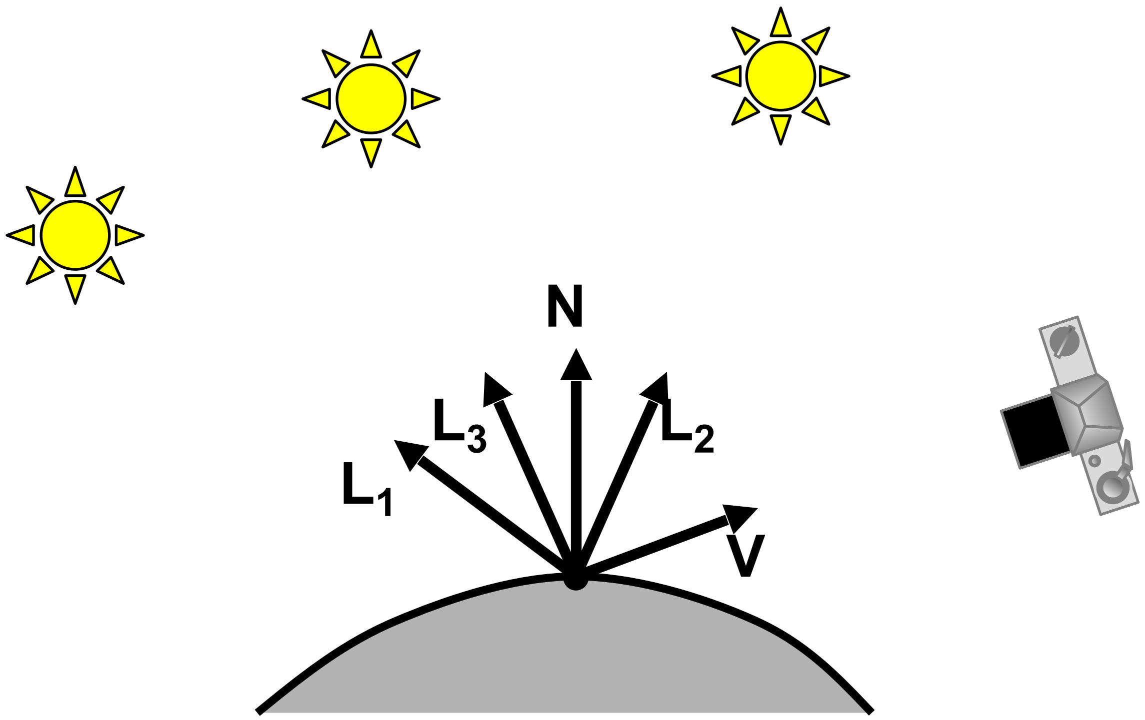
Assume $k_d$ is 1 for now.

What can we measure from one image?

- $\cos^{-1}(I)$ is the angle between N and L
- Add assumptions:
  - A few known normals (e.g. silhouettes)
  - Smoothness of normals

In practice, SFS doesn't work very well: assumptions are too restrictive, too much ambiguity in nontrivial scenes.

# Multiple Images: Photometric Stereo




$$I_1 = k_d \mathbf{N} \cdot \mathbf{L_1}$$
$$I_2 = k_d \mathbf{N} \cdot \mathbf{L_2}$$
$$I_3 = k_d \mathbf{N} \cdot \mathbf{L_3}$$

Write this as a matrix equation:

$$\left[ \begin{array}{ccc} I_1 & I_2 & I_3 \end{array} \right] = k_d \mathbf{N}^T \left[ \begin{array}{ccc} \mathbf{L_1} & \mathbf{L_2} & \mathbf{L_3} \end{array} \right]$$

# Solving the Equations

$$\underbrace{\begin{bmatrix} I_1 & I_2 & I_3 \end{bmatrix}}_{\substack{\mathbf{I} \\ 1\times 3}} = \underbrace{k_d \mathbf{N}^T}_{\substack{\mathbf{G} \\ 1\times 3}} \underbrace{\begin{bmatrix} \mathbf{L_1} & \mathbf{L_2} & \mathbf{L_3} \end{bmatrix}}_{\substack{\mathcal{L} \\ 3\times 3}}$$

$$\mathbf{G} = \mathbf{I}\mathbf{L}^{-1}$$

$$k_d = \|\mathbf{G}\|$$

$$\mathbf{N} = \frac{1}{k_d}\mathbf{G}$$

# Solving the Equations

$$\underbrace{\begin{bmatrix} I_1 & I_2 & I_3 \end{bmatrix}}_{\substack{\mathbf{I} \\ 1\times 3}} = \underbrace{k_d \mathbf{N}^T}_{\substack{\mathbf{G} \\ 1\times 3}} \underbrace{\begin{bmatrix} \mathbf{L_1} & \mathbf{L_2} & \mathbf{L_3} \end{bmatrix}}_{\substack{\mathcal{L} \\ 3\times 3}}$$

$$\mathbf{G} = \mathbf{I}\mathbf{L}^{-1}$$

- When is L nonsingular (invertible)?
  - >= 3 light directions are linearly independent, or:
  - All light direction vectors cannot lie in a plane.
- What if we have more than one pixel?
  - Stack them all into one big system.

# More than Three Lights

$$\left[ \begin{array}{ccc} I_1 & \dots & I_n \end{array} \right] = k_d \mathbf{N}^T \left[ \begin{array}{ccc} \mathbf{L_1} & \dots & \mathbf{L_n} \end{array} \right]$$

- Solve using least squares (normal equations):

$$\begin{aligned} \mathbf{I} &= \mathbf{GL} \\ \mathbf{IL^T} &= \mathbf{GLL^T} \\ \mathbf{G} &= (\mathbf{IL^T})(\mathbf{LL^T})^{-1} \end{aligned}$$

- Or equivalently, use the SVD.
- Given G, solve for N and $k_d$ as before.

# More than one pixel

Previously:

1 x # images | 1 x 3 | 3 x # images

I = N * L

# More than one pixel

Stack all pixels into one system:

| p x # images | | p x 3 | | 3 x # images |
|---|---|---|---|---|
| I | = | N | * | L |

Solve as before.

# Color Images

- Now we have 3 equations for a pixel:

$$\begin{aligned} \mathbf{I}_R &= k_{dR}\,\mathbf{LN} \\ \mathbf{I}_G &= k_{dG}\,\mathbf{LN} \\ \mathbf{I}_B &= k_{dB}\,\mathbf{LN} \end{aligned}$$

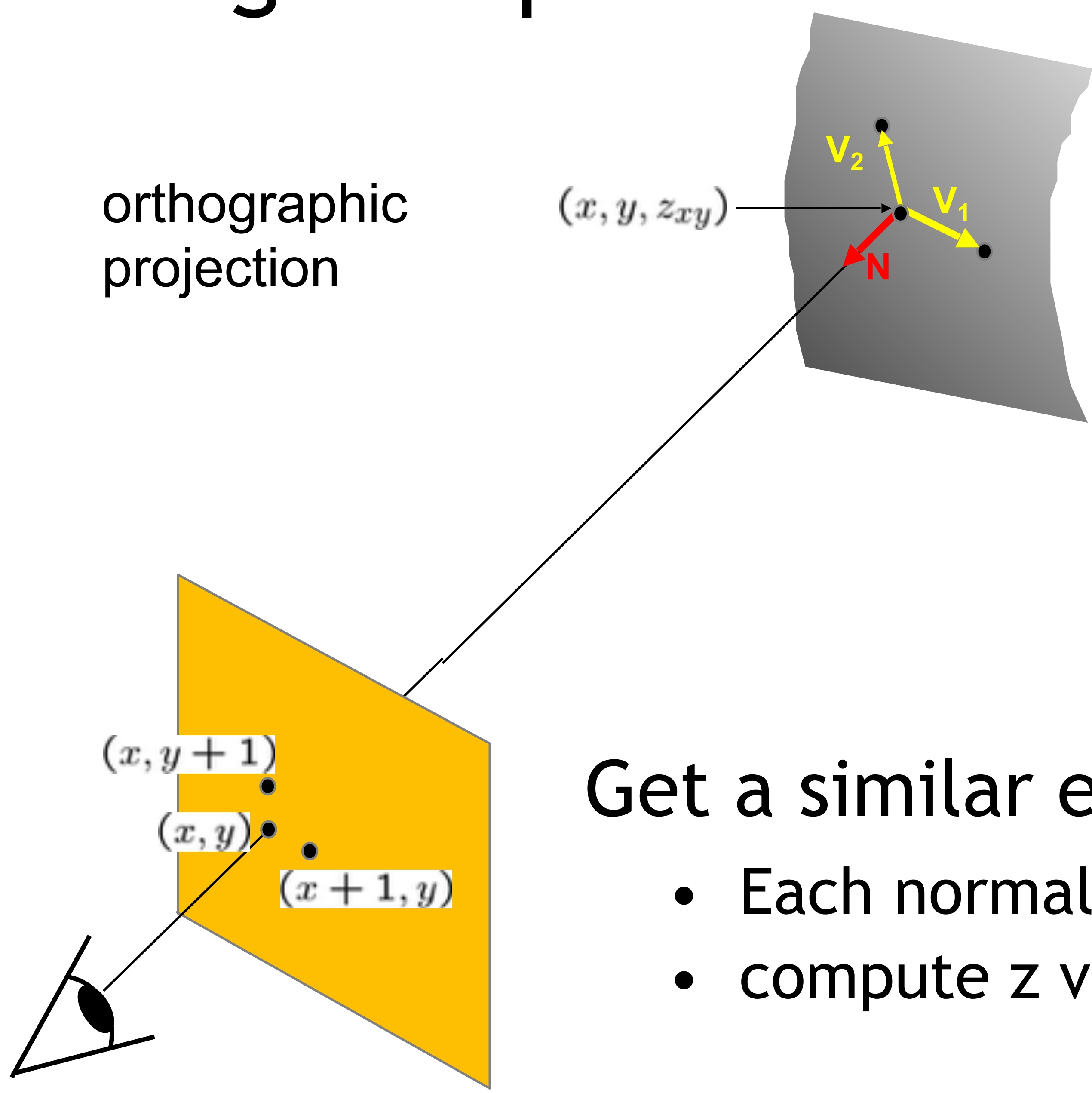
- Simple approach: solve for N using grayscale or a single channel.
- Then fix N and solve for each channel's $k_d$:

$$k_d = \frac{\sum_i I_i L_i N^T}{\sum_i (L_i N^T)^2}$$

# Depth Map from Normal Map

- We now have a surface normal, but how do we get depth?

Assume a smooth surface




$$
\begin{aligned}
V_1 &= (x+1, y, z_{x+1,y}) - (x, y, z_{xy}) \\
&= (1, 0, z_{x+1,y} - z_{xy})
\end{aligned}
$$

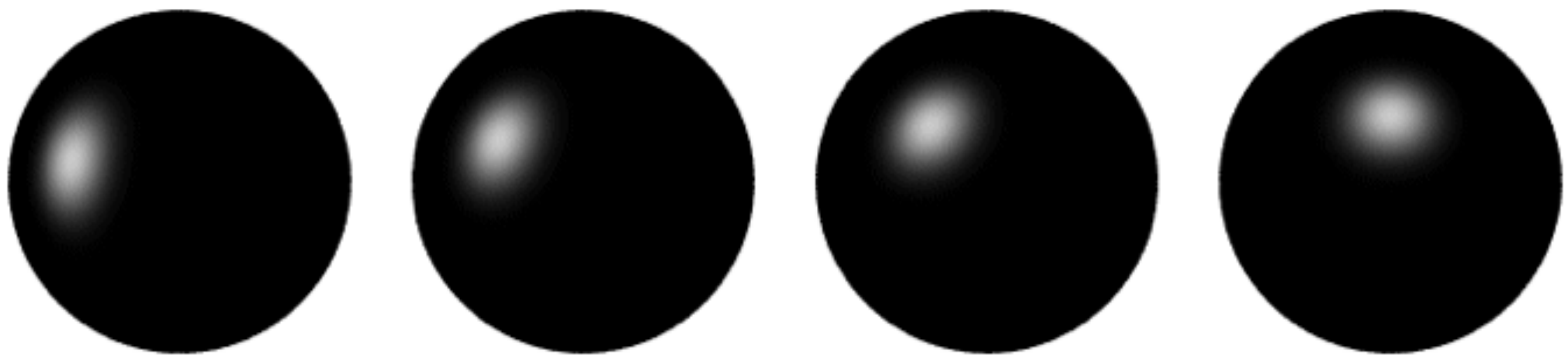
$$
\begin{aligned}
0 &= N \cdot V_1 \\
&= (n_x, n_y, n_z) \cdot (1, 0, z_{x+1,y} - z_{xy}) \\
&= n_x + n_z(z_{x+1,y} - z_{xy})
\end{aligned}
$$

Get a similar equation for $\mathbf{V_2}$

- Each normal gives us two linear constraints on z
- compute z values by solving a matrix equation

# Determining Light Directions

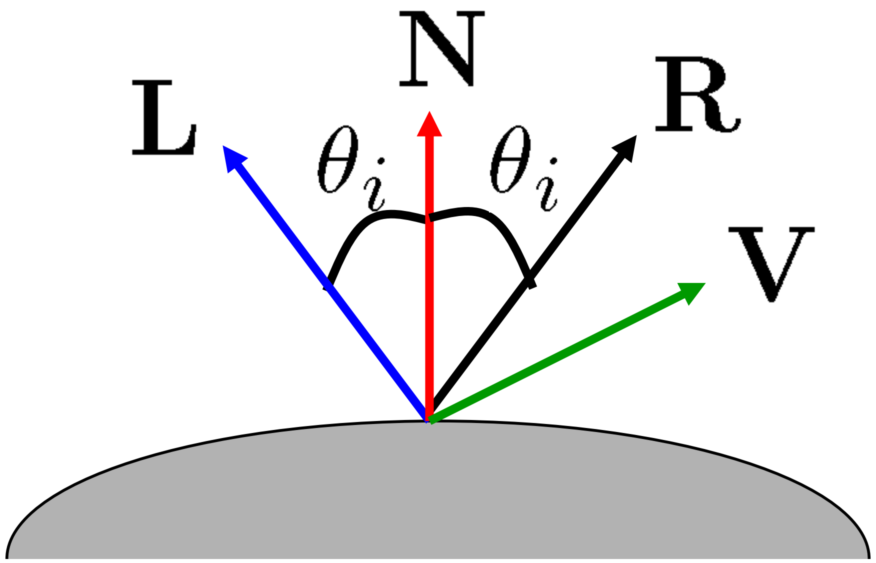
- Trick: Place a mirror ball in the scene.



- The location of the highlight is determined by the light source direction.

# Determining Light Directions

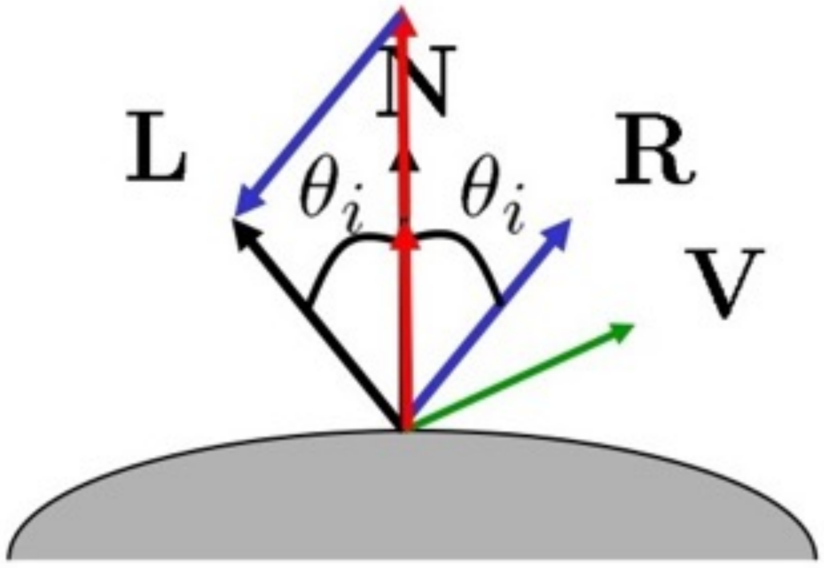
- For a perfect mirror, the light is reflected across N:




$$I_e = \begin{cases} I_i & \text{if } \mathbf{V} = \mathbf{R} \\ 0 & \text{otherwise} \end{cases}$$

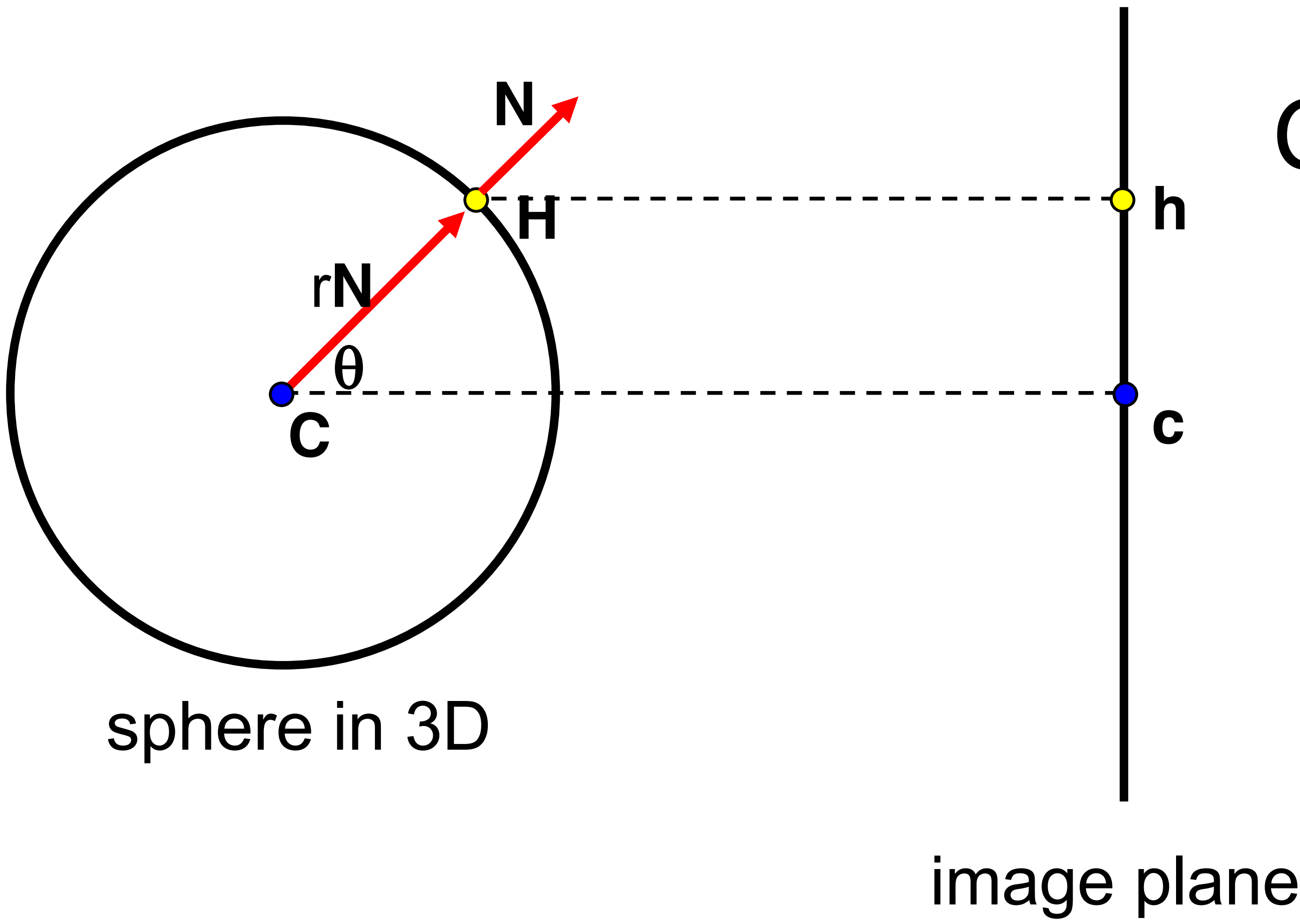
- So the light source direction is given by:

$$L = 2(N \cdot R)N - R$$



# Determining Light Directions

- For a sphere with highlight at point H:




Compute N:

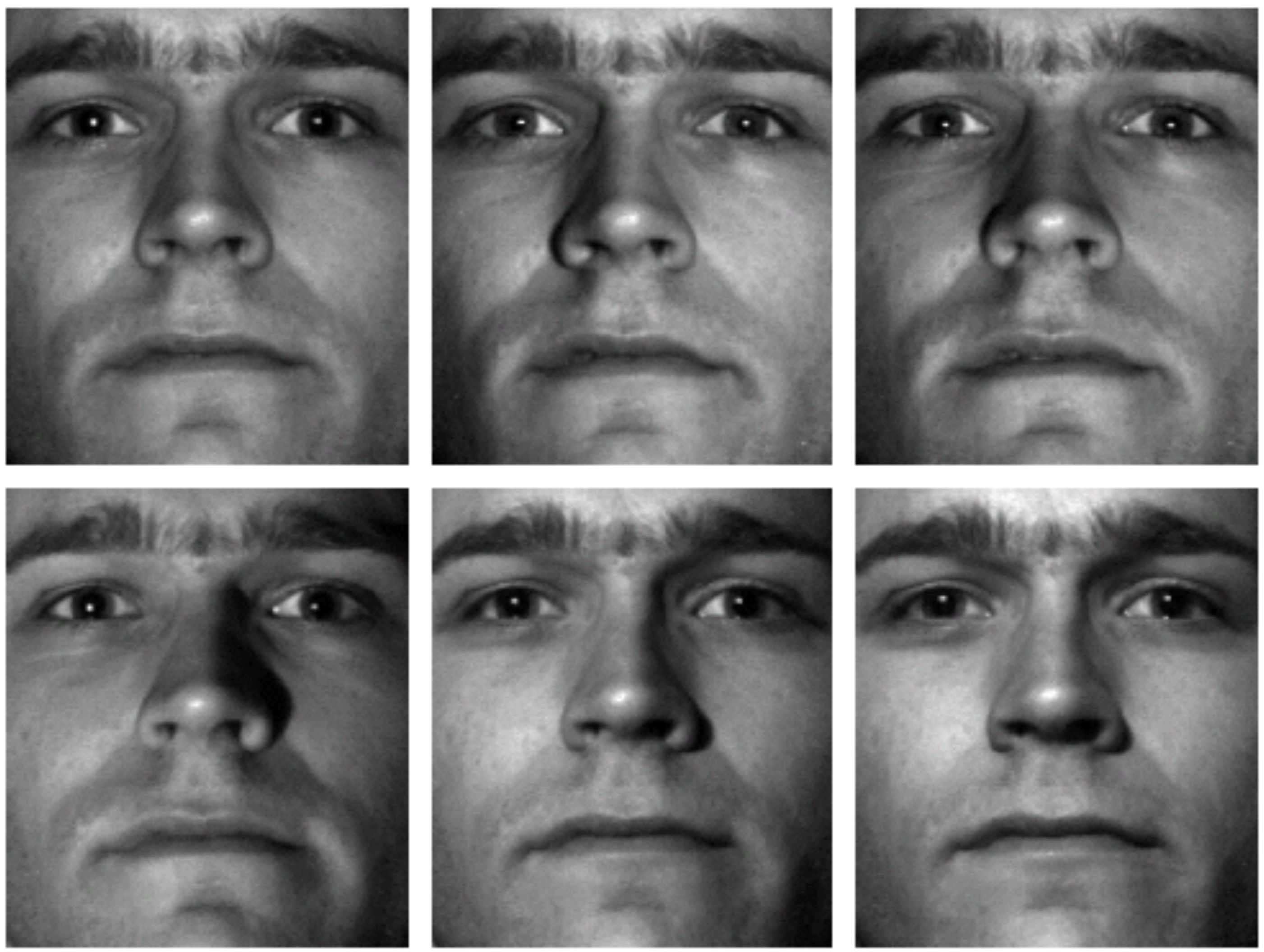
$$N_x = \frac{x_h - x_c}{r}$$

$$N_y = \frac{y_h - y_c}{r}$$

$$N_z = \sqrt{1 - x^2 - y^2}$$

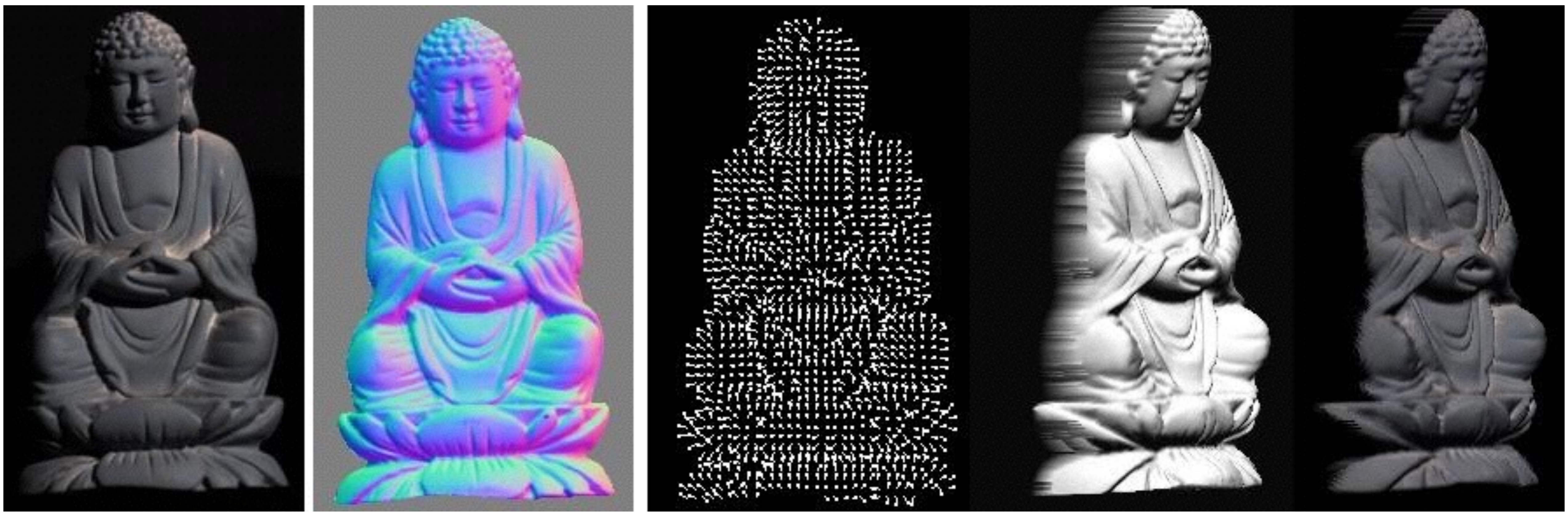
- R = direction of the camera from C = $[0\ 0\ 1]^T$
  $L = 2(N \cdot R)N - R$

# Results



from Athos Georghiades

# Results



Input (1 of 12) | Normals (RGB colormap) | Normals (vectors) | Shaded 3D rendering | Textured 3D rendering

# For (unfair) Comparison

- Multi-view stereo results on a similar object
- 47+ hrs compute time

State-of-the-art MVS result

Ground truth

# Taking Stock: Assumptions

| Lighting | Materials | Geometry | Camera |
|---|---|---|---|
| directional | diffuse | convex / no shadows | linear |
| known direction | no inter-reflections | | orthographic |
| > 2 nonplanar directions | no subsurface scattering | | |

Questions?

# Unknown Lighting

- What we’ve seen so far: [Woodham 1980]
- Next up: Unknown light directions [Hayakawa 1994]

# Unknown Lighting

Surface normals

Light directions

$$I = kN \cdot \ell L$$

Diffuse albedo

Light intensity

# Unknown Lighting

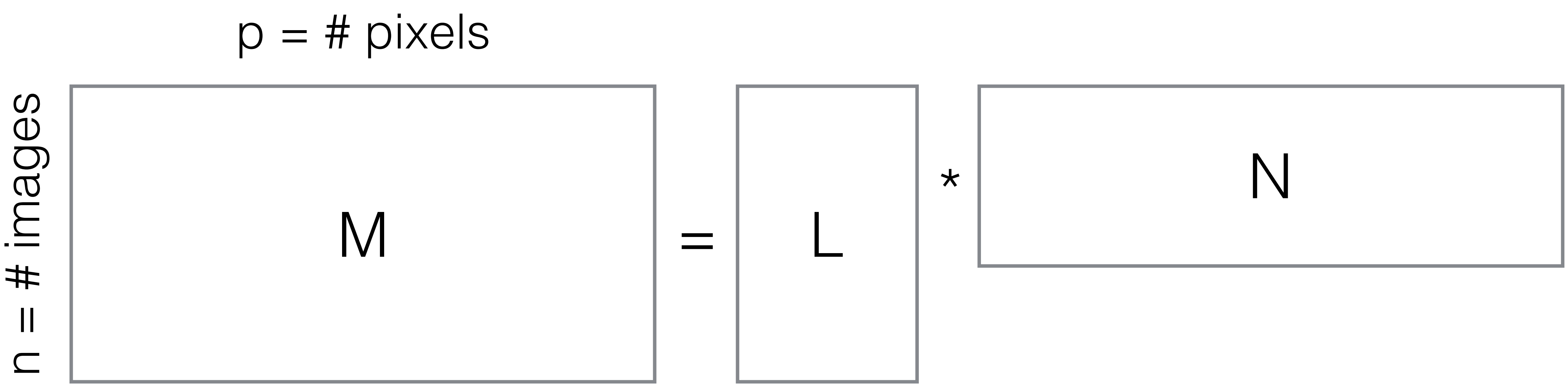
Surface normals, scaled by albedo

Light directions, scaled by intensity

$$I = N \cdot L$$

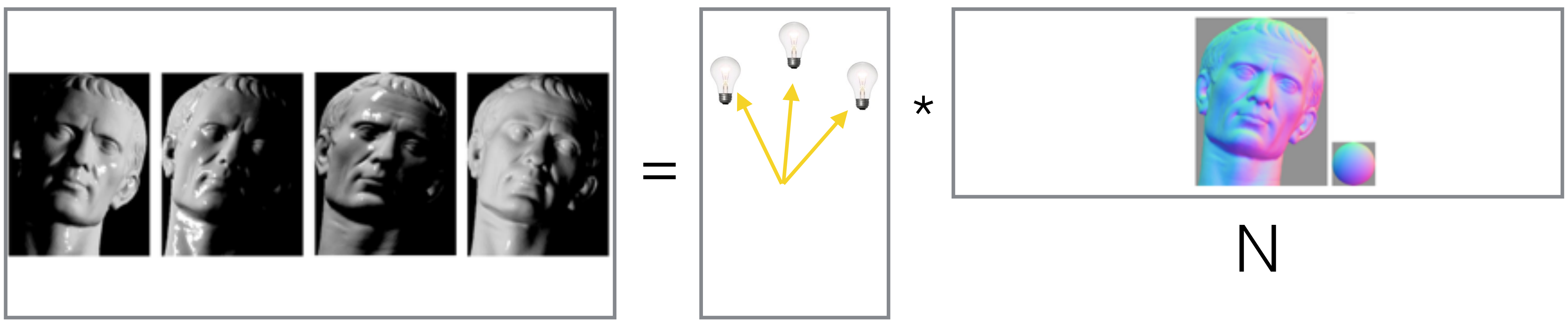
# Unknown Lighting

Same as before, just transposed:



# Unknown Lighting

Measurements (one image per row) | Light directions (scaled by intensity) | Surface normals (scaled by albedo)

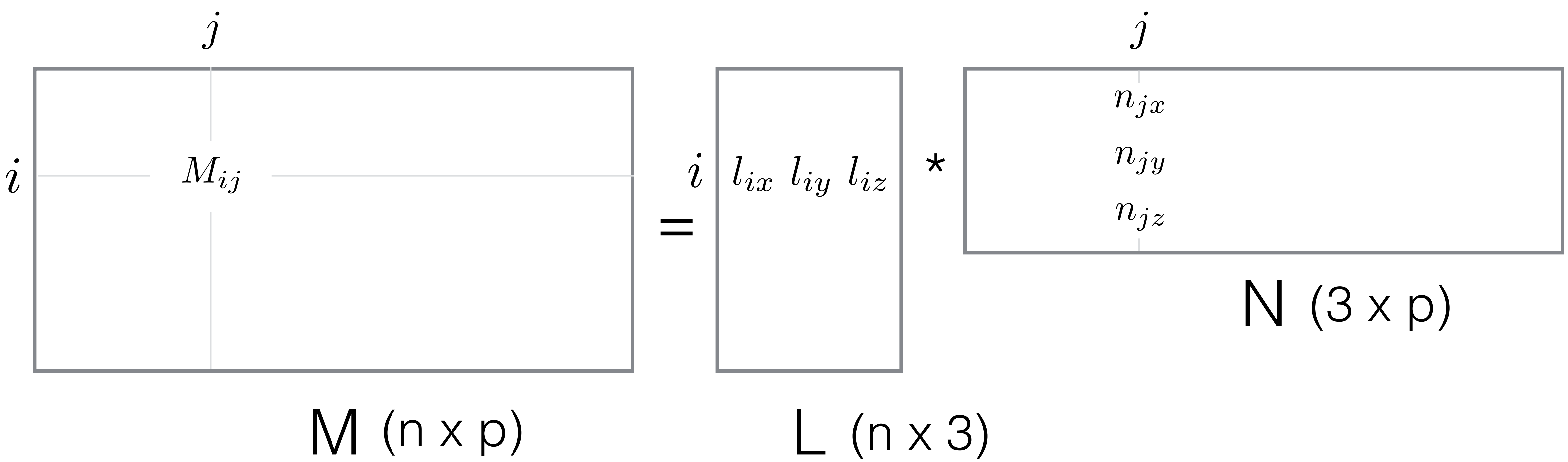



M L

**Both L and N are now unknown!**
**This is a matrix factorization problem.**

# Unknown Lighting

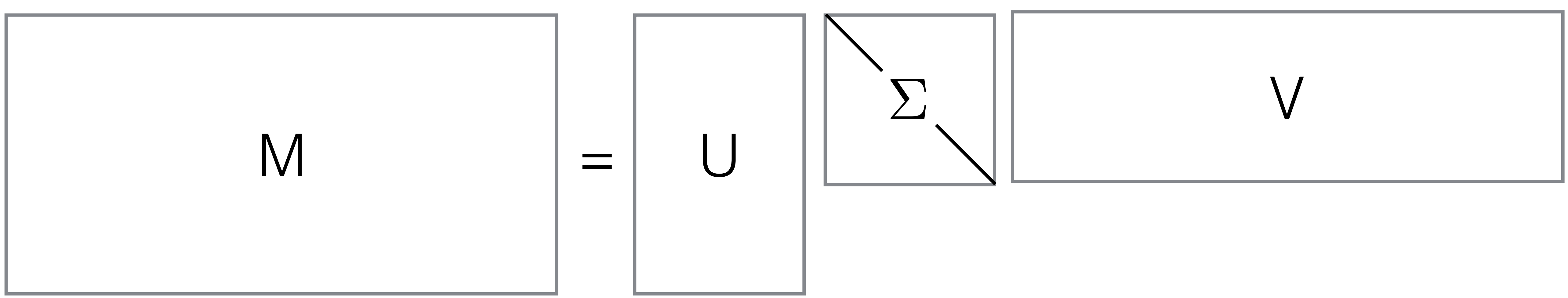
$$M_{ij} = L_i \cdot N_j$$




There's hope: We know that M is rank 3

# Unknown Lighting

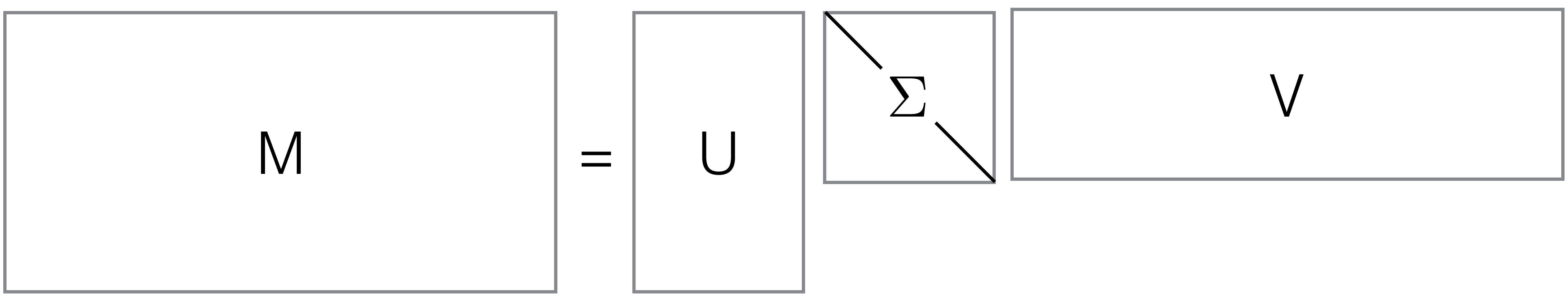
Use the SVD to decompose M:




SVD gives the best rank-3 approximation of a matrix.

# Unknown Lighting

Use the SVD to decompose M:




What do we do with $\Sigma$?

# Unknown Lighting

Use the SVD to decompose M:

$$M = U\sqrt{\Sigma}\ \sqrt{\Sigma}V$$

What do we do with $\Sigma$?

# Unknown Lighting

Use the SVD to decompose M:

$$M = U\sqrt{\Sigma}\ \sqrt{\Sigma}V$$

Can we just do that?

# Unknown Lighting

Use the SVD to decompose M:

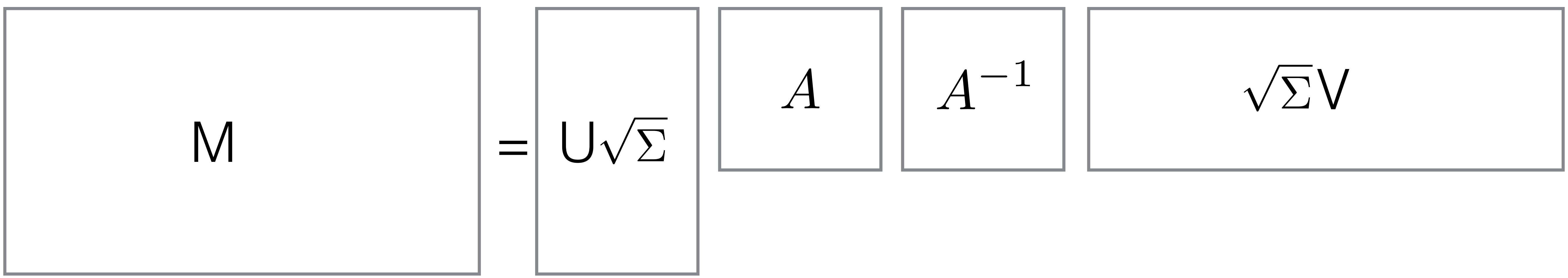
$$M = U\sqrt{\Sigma} \; A \; A^{-1} \; \sqrt{\Sigma}V$$

Can we just do that? …almost.

The decomposition is non-unique up to an invertible 3x3 A.

# Unknown Lighting

Use the SVD to decompose M:




$$L = U\sqrt{\Sigma}A \qquad S = A^{-1}\sqrt{\Sigma}V$$

# Unknown Lighting

Use the SVD to decompose M:

$$M = U\sqrt{\Sigma} \; A \; A^{-1} \; \sqrt{\Sigma}V$$

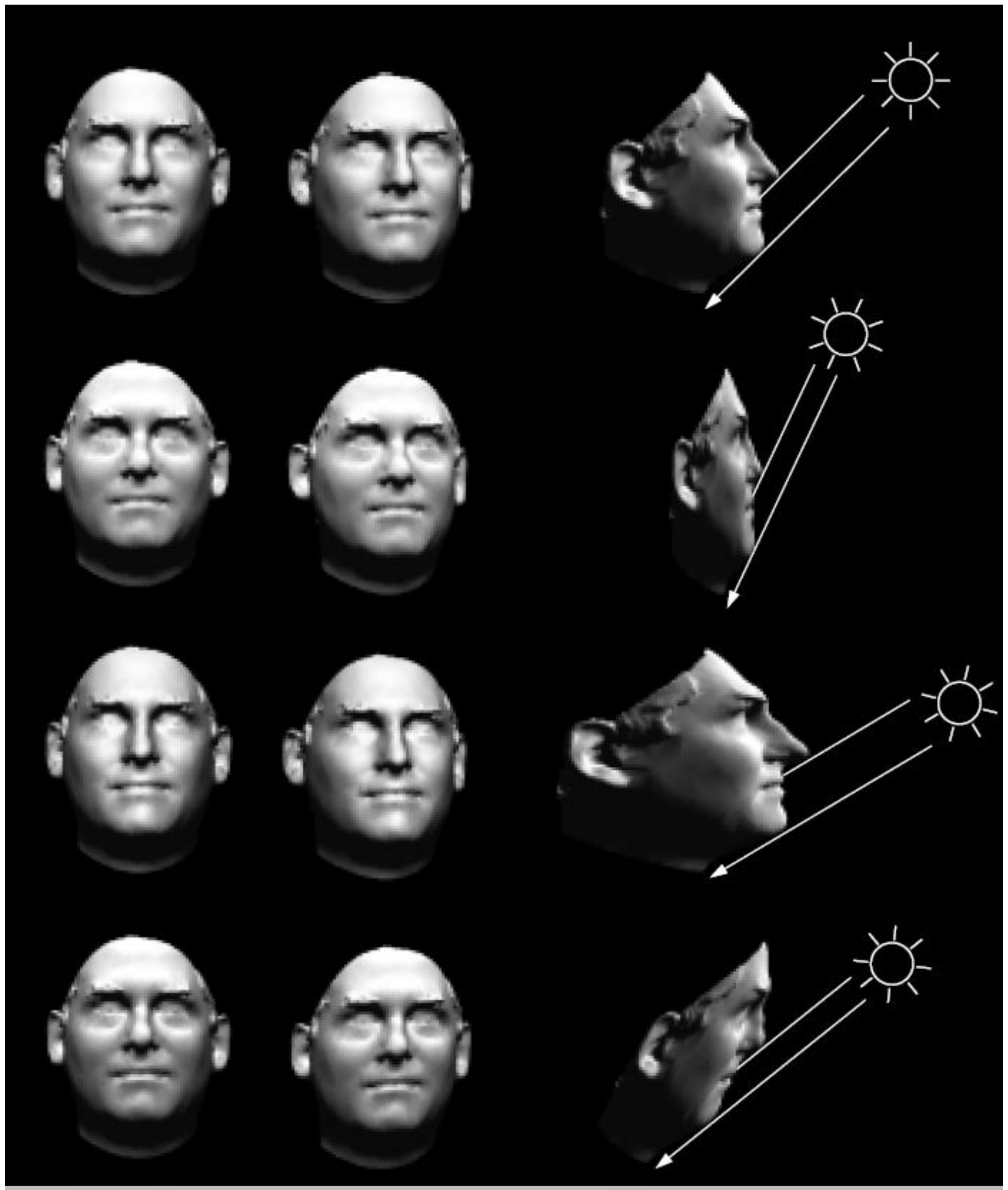
$$L = U\sqrt{\Sigma}A \qquad S = A^{-1}\sqrt{\Sigma}V$$

You can find A if you know

- 6 points with the same reflectance, or
- 6 lights with the same intensity.

# Unknown Lighting: Ambiguities

- Multiple combinations of lighting and geometry can produce the same sets of images.
- Add assumptions or prior knowledge about geometry or lighting, etc. to limit the ambiguity.

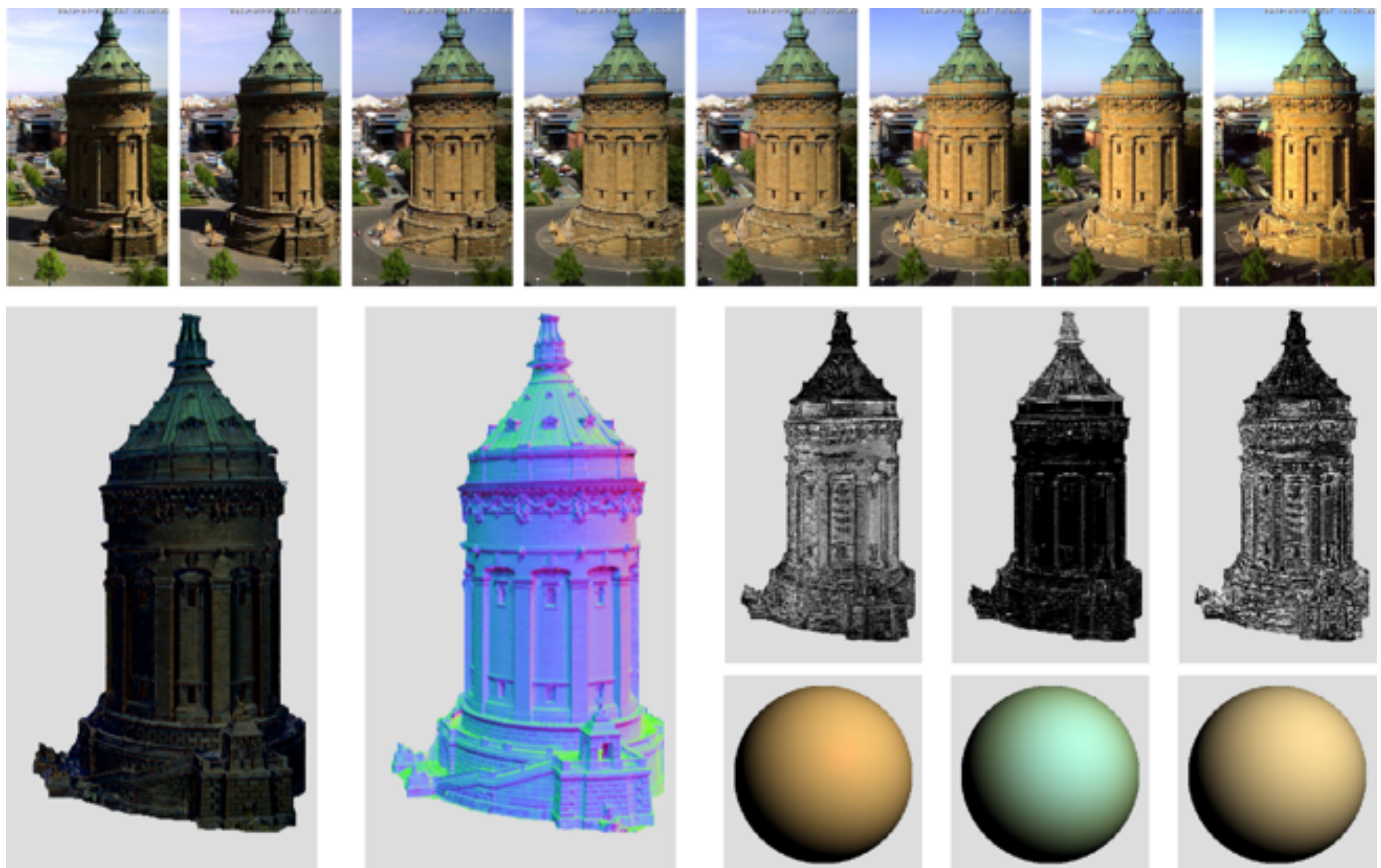


[Belhumeur et al.'97]

Questions?

# Since 1994…

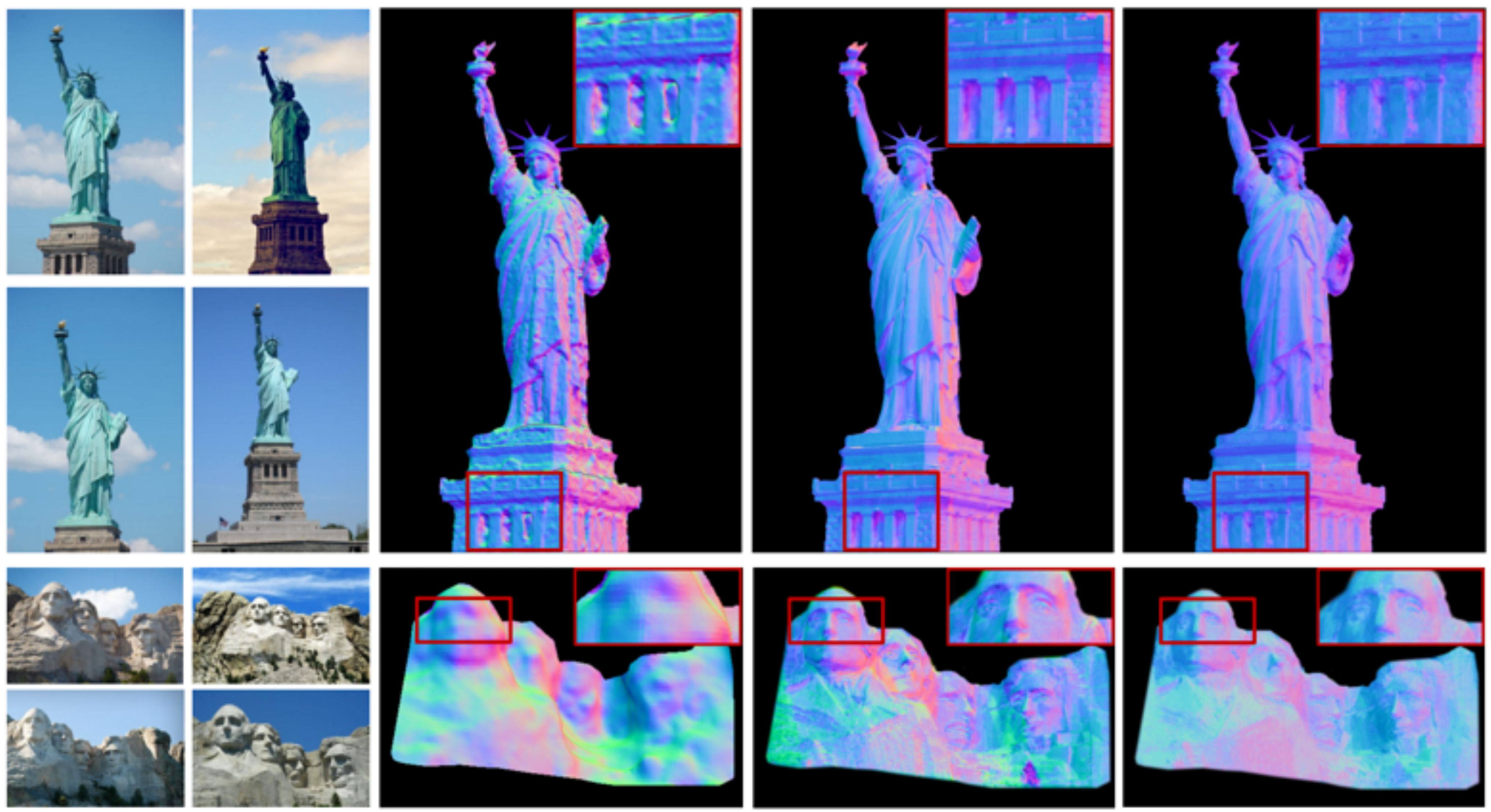
- Workarounds for many of the restrictive assumptions.
- Webcam photometric stereo:



Ackermann et al. 2012

# Since 1994...

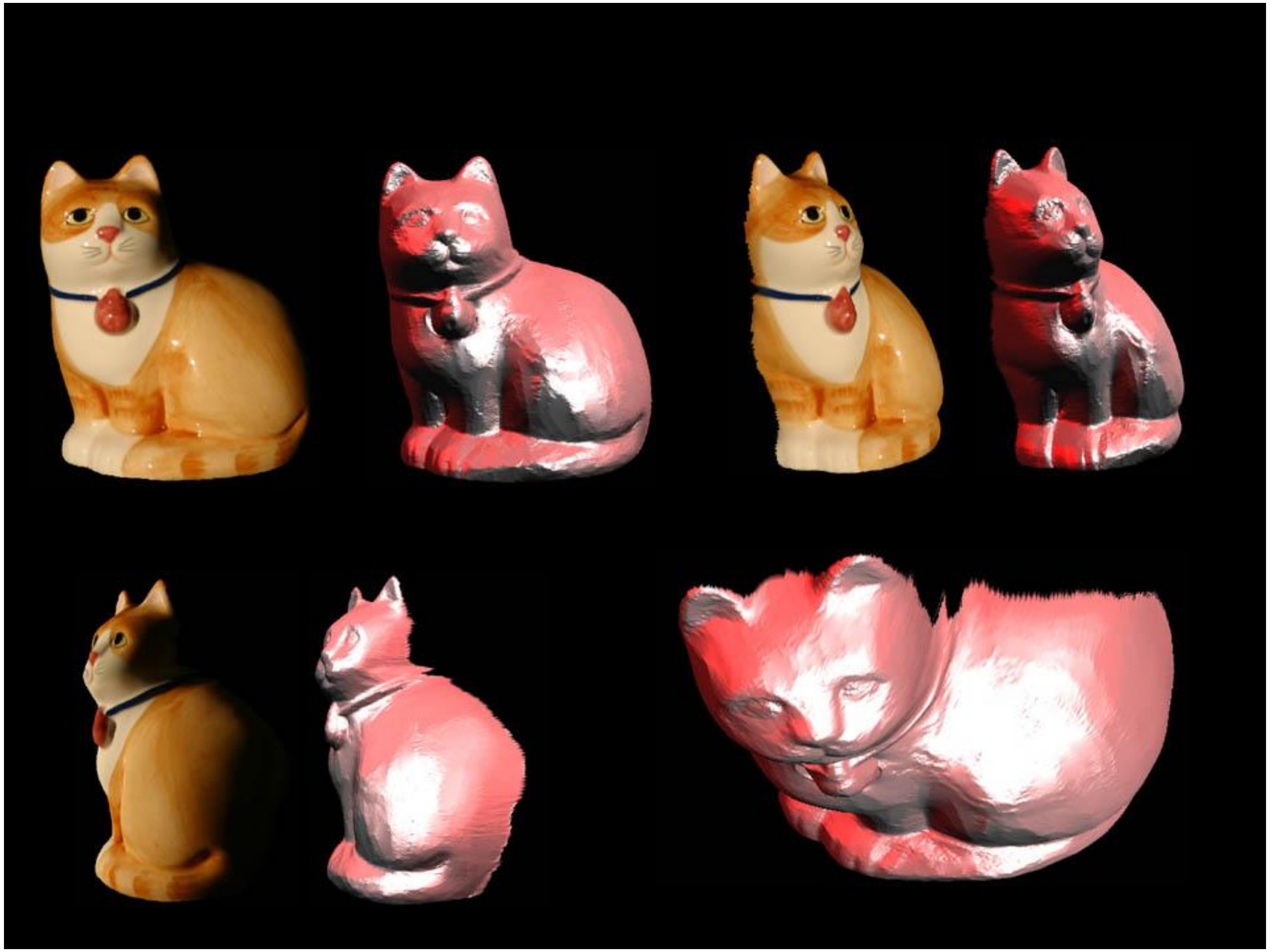
- Photometric stereo from unstructured photo collections (different cameras and viewpoints):



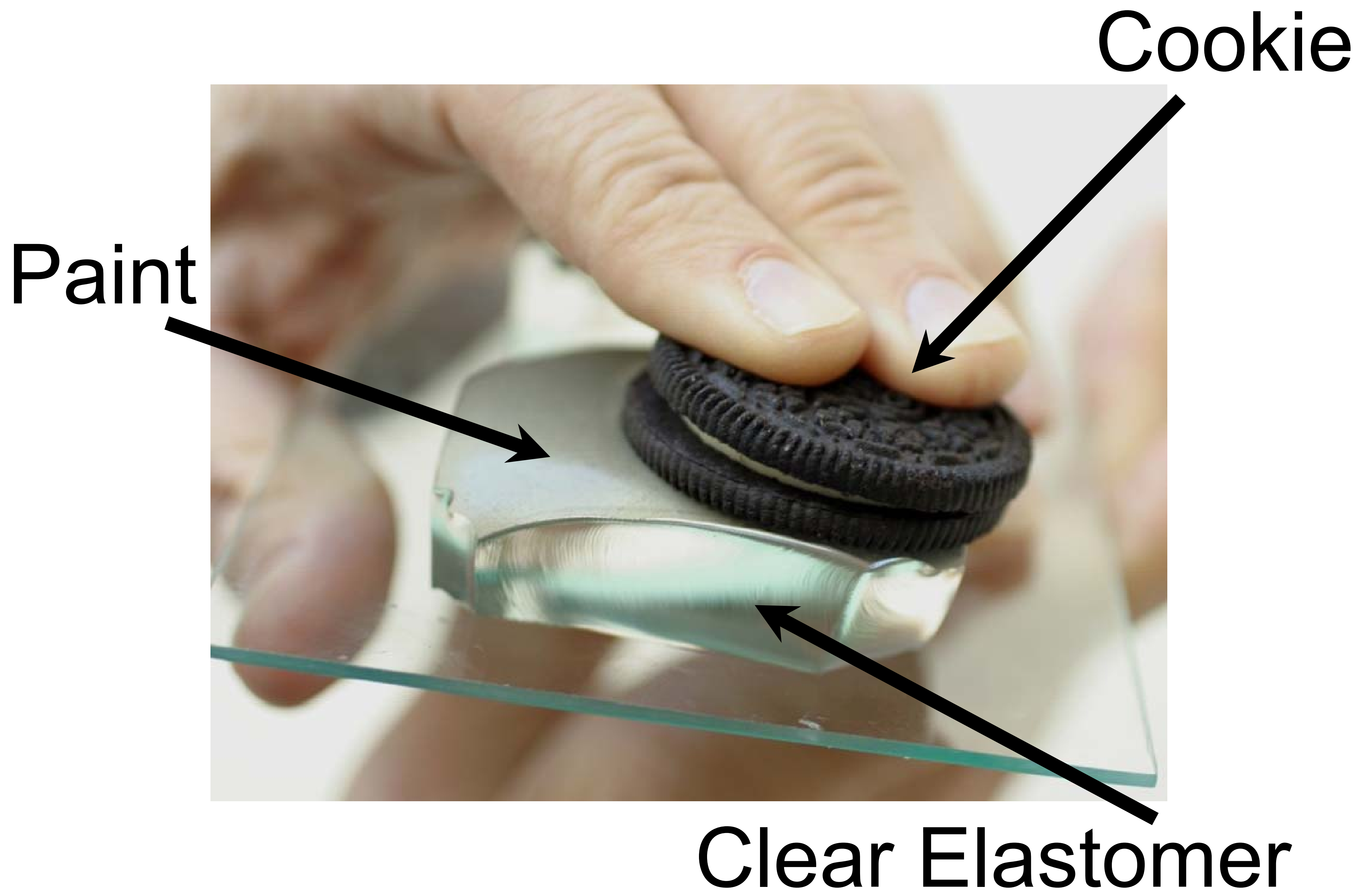
Shi et al, 2014

# Since 1994…

- Non-Lambertian (shiny) materials:



Hertzmann and Seitz, 2005


Cookie
Paint
Clear Elastomer


OREO


OREO

# Lights, camera, action