Photometric Stereo Recap

Introduction to Computer Vision
CSE152
Lecture 16

Announcement
An interesting analysis of Midterm Scores of those who attend class: 65.6%
Scores of those who skip class: 42.7%

• HW3 posted – Photometric stereo
• Discussion Section
  – Thursday 5/19
  – Time: 1-1:50 pm
  – Location: SERF building, room 102

Photometric Stereo Rigs:
One viewpoint, changing lighting

Camera stays fixed, scene is static.
One light turned on and first image is acquired
Second light is turned on, and second image is acquired, etc.
Surface normals are estimated and then surface is integrated.

An example of photometric stereo

BRDF
• Bi-directional Reflectance Distribution Function
  \( \rho(\theta_{\text{in}}, \phi_{\text{in}}; \theta_{\text{out}}, \phi_{\text{out}}) \)
• Function of
  \( \theta_{\text{in}} \) and \( \phi_{\text{in}} \)
  \( \theta_{\text{out}} \) and \( \phi_{\text{out}} \)
• Ratio of incident irradiance to emitted radiance

Photometric Stereo: Three problems
1. General but known reflectance function
2. Lambertian surfaces with known lighting
3. Lambertian surfaces with unknown lighting
Photometric Stereo:
General BRDF and Reflectance Map

Coordinate system

Surface: \( s(x,y) = (x, y, f(x,y)) \)
Tangent vectors:
\[
\begin{align*}
\frac{\partial s(x,y)}{\partial x} &= \begin{bmatrix} 1 \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix} \\
\frac{\partial s(x,y)}{\partial y} &= \begin{bmatrix} 0 \frac{\partial f}{\partial x} \\ 1 \frac{\partial f}{\partial y} \end{bmatrix}
\end{align*}
\]
Normal vector
\[
\mathbf{n} = \frac{\partial s(x,y)}{\partial x} \times \frac{\partial s(x,y)}{\partial y} = \begin{bmatrix} \frac{\partial f}{\partial y} & -\frac{\partial f}{\partial x} & 0 \end{bmatrix}
\]

Gradient Space (p,q)

Image Formation

For a given point A on the surface, the image irradiance \( E(x,y) \) is a function of
1. The BRDF at A
2. The surface normal at A
3. The direction of the light source

Reflectance Map

Let the BRDF be the same at all points on the surface, and let the light direction \( s \) be constant.
1. Then image irradiance is a function of only the direction of the surface normal.
2. In gradient space, we can write \( E(p,q) \)
3. We can measure \( E(p,q) \) by taking an image of a sphere made of a single material under distant lighting

Example Reflectance Map: Lambertian surface

For lighting from front
What does the intensity (irradiance) of one pixel in one image tell us?
It constrains the surface normal projecting to that point to a curve.

E.g., Normal lies on this curve.

Two Light Sources
Two reflectance maps

A third image would disambiguate match.

Three Source Photometric stereo:
Step 1
Offline:
Using source directions & BRDF, construct reflectance map
for each light source direction. \( R_1(p,q), R_2(p,q), R_3(p,q) \)
Online:
1. Acquire three images with known light source directions.
   \( E_1(x,y), E_2(x,y), E_3(x,y) \)
2. For each pixel location \((x,y)\), find \((p,q)\) as the intersection
   of the three curves
   \( R_1(p,q) = E_1(x,y) \)
   \( R_2(p,q) = E_2(x,y) \)
   \( R_3(p,q) = E_3(x,y) \)
3. This is the surface normal at pixel \((x,y)\). Over image, the
   normal field is estimated.

An Example Normal Field

Plastic Baby Doll: Normal Field
Next step:
Go from normal field to surface

Recovering the surface \( f(x,y) \)
Many methods: Simplest approach
1. From normal field \( n = (n_x, n_y, n_z) \), \( p = -n_x/n_z \), \( q = -n_y/n_z \)
2. Integrate \( p = df/dx \) along a row \((x,0)\) to get \( f(x,0) \)
3. Then integrate \( q = df/dy \) along each column starting with value of the first row

II. Photometric Stereo:
Lambertian Surface, Known Lighting

Lambertian Photometric stereo
• If the light sources \( s_1, s_2, \) and \( s_3 \) are known, then we can recover \( b \) from as few as three images. (Photometric Stereo: Silver 80, Woodham81).

\[
\begin{bmatrix} e_1 & e_2 & e_3 \end{bmatrix} = b^T \begin{bmatrix} s_1 & s_2 & s_3 \end{bmatrix}
\]
• i.e., we measure \( e_1, e_2, \) and \( e_3 \) and we know \( s_1, s_2, \) and \( s_3 \). We can then solve for \( b \) by solving a linear system.

\[
b^T = \begin{bmatrix} e_1 & e_2 & e_3 \end{bmatrix} \begin{bmatrix} s_1 & s_2 & s_3 \end{bmatrix}^{-1}
\]
• Normal is: \( n = b/|b| \), albedo is: \(|b|\)

What if we have more than 3 Images?
Linear Least Squares
Let the residual be \( r = e - Sb \)
Rewrite as \( e = Sb \)
Squaring this:
\[
r^T = r^T (e - Sb)^T (e - Sb) = e^T e - 2e^T S b + S^T S b
\]
where
\[
e \text{ is n by 1}
\]
\[
b \text{ is 3 by 1}
\]
\[
S \text{ is n by 3}
\]
\[
V_e(r^2) = 0 \text{ - zero derivative is a necessary condition for a minimum, or}
\]
\[
-2S^T S b = 0;
\]
Solving for \( b \) gives
\[
b = (S^T S)^{-1} S^T e
\]
**Input Images**

![Input Images](image)

**Recovered albedo**

![Recovered albedo](image)

**Recovered normal field**

![Recovered normal field](image)

**Surface recovered by integration**

![Surface recovered by integration](image)

**Lambertian Photometric Stereo**

![Lambertian Photometric Stereo](image)

**Reconstruction with albedo map**

![Reconstruction with albedo map](image)
Without the albedo map

Another person

III. Photometric Stereo with unknown lighting and Lambertian surfaces

Generalized Bas-Relief Transformations

Objects differing by a GBR have the same illumination cone.

Without knowledge of light source location, one can only recover surfaces up to GBR transformations.

GBR Transformation

Only Generalized Bas-Relief transformations satisfy the integrability constraint:

\[ A = G^T = \begin{bmatrix} \lambda & 0 & -\mu \\ 0 & \lambda & -v \\ 0 & 0 & 1 \end{bmatrix} \]

Uncalibrated photometric stereo

1. Take n images as input, perform SVD to compute \( B^* \).
2. Find some \( A \) such that \( B^*A \) is close to integrable.
3. Integrate resulting gradient field to obtain height function \( f^*(x,y) \).

Comments:
- \( f^*(x,y) \) differs from \( f(x,y) \) by a GBR.
- Can use specularities to resolve GBR for non-Lambertian surface.
Announcement

• HW3 posted – Photometric stereo

• Discussion Section
  – Thursday 5/19
  – Time: 1:15:50 pm
  – Location: SERF building, room 102

Family of epipolar Planes

Family of planes that intersect at the baseline.
Family of lines $l$ and $l'$ that intersect in $e$ and $e'$

The Fundamental matrix

The epipolar constraint is given by: $p^T E p' = 0$ with $E = \{ e \}_R$

where $p$ and $p'$ are 3-D coordinates of the image coordinates of points in the two images.

Without calibration, we can still identify corresponding points in two images, but we can’t convert to 3-D coordinates. However, the relationship between the calibrated coordinates $(p,p')$ and uncalibrated coordinates $(q,q')$ can be expressed as $p = Aq$, and $p' = A'q'$. Therefore, we can express the epipolar constraint as:

$$ (Aq)^T E (A'q') = q^T (A^T EA') q' = q^T F q' = 0 $$

where $F$ is called the Fundamental Matrix.
The Eight-Point Algorithm (Longuet-Higgins, 1981)

\[ (u, v) \begin{bmatrix} F_{11} & F_{12} & F_{13} & u' \end{bmatrix} \begin{bmatrix} F_{21} & F_{22} & F_{23} & v' \end{bmatrix} \begin{bmatrix} 1 \end{bmatrix} = 0 \]

Set \( F_{33} = 1 \)

Same as minimizing

\[ \sum \frac{1}{n} (p_i^t \cdot F \cdot p_i) \]

under the constraint

\[ \|F\|_1 = 1 \]

Rectification

Given a pair of images, transform both images so that epipolar lines are scan lines.

Image pair rectification

simplify stereo matching by warping the images

Apply projective transformation \( H \) so that epipolar lines correspond to horizontal scanlines

Mobi: Stereo-based navigation
Epipolar correspondence

This version is feature-based: detect edges in 1-D signal, and use dynamic programming to find correspondences that minimize an error function.

Using epipolar & constant Brightness constraints for stereo matching

For each epipolar line
For each pixel in the left image
• compare with every pixel on same epipolar line in right image
• pick pixel with minimum match cost
• This will never work, so:

Improvement: match windows
(Seitz)

Comparing Windows:

\[
SSD = \sum_{i,j \in R} (f(i,j) - g(i,j))^2
\]

\[
Cf_g = \sum_{i,j \in R} f(i,j)g(i,j)
\]

Most popular

For each window, match to closest window on epipolar line in other image.

(Camps)

Symbolic Map

Finding Correspondences

Correspondence Search Algorithm

\[
\text{for } k = \text{mindisparity:maxdisparity}
\]
\[
c = \text{Match Metric}(i,j,l_1(i,j+k),\text{winsize})
\]
\[
\text{if } (c > \text{best}(i,j))
\]
\[
\text{best}(i,j) = c
\]
\[
\text{disparities}(i,j) = k
\]
\[
\text{end}
\]
\[
\text{end}
\]
\[
O(nrows * ncols * disparities * winx * winy)
\]
<table>
<thead>
<tr>
<th>MATCH METRIC</th>
<th>DEFINITION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Normalized Cross-Correlation (NCC)</td>
<td>( \frac{\sum_{i,j} (A_{i,j} - \mu_A)(B_{i,j} - \mu_B)}{\sqrt{\sum_{i,j} (A_{i,j} - \mu_A)^2 \sum_{i,j} (B_{i,j} - \mu_B)^2}} )</td>
</tr>
<tr>
<td>Sum of Squared Differences (SSD)</td>
<td>( \sum_{i,j} (A_{i,j} - B_{i,j})^2 )</td>
</tr>
<tr>
<td>Normalized SSD</td>
<td>( \frac{\sum_{i,j} (A_{i,j} - B_{i,j})^2}{\sqrt{\sum_{i,j} A_{i,j}^2 \sum_{i,j} B_{i,j}^2}} )</td>
</tr>
<tr>
<td>Sum of Absolute Differences (SAD)</td>
<td>( \sum_{i,j}</td>
</tr>
<tr>
<td>Zero Mean SAD</td>
<td>( \sum_{i,j}</td>
</tr>
<tr>
<td>Rank</td>
<td>( \frac{\sum_{i,j} \frac{A_{i,j}}{B_{i,j}}}{\sum_{i,j} B_{i,j}} )</td>
</tr>
<tr>
<td>Census</td>
<td>( \frac{\sum_{i,j} \text{match}(A_{i,j}, B_{i,j})}{\sum_{i,j} \text{match}(A_{i,j}, B_{i,j})} )</td>
</tr>
</tbody>
</table>

These two are actually the same