6.801/6.866: Machine Vision, Quiz 1 Review

Professor Berthold Horn, Ryan Sander, Tadayuki Yoshitake
MIT Department of Electrical Engineering and Computer Science
Fall 2020

These lecture summaries are designed to be a review of the lecture. Though I do my best to include all main topics from the lecture, the lectures will have more elaborated explanations than these notes.

Here you will find a review of some of the topics covered so far in Machine Vision. These are as follows in the section notes:

1. **Mathematics review** - Unconstrained optimization, Green's Theorem, Bezout's Theorem, Nyquist Sampling Theorem

2. **Projection** - Perspective vs. Orthographic, Object vs. Image space

3. **Optical Flow** - BCCE/optical flow, Time to Contact, vanishing points

4. **Photometry** - Radiance, irradiance, illumination, geometry, BRDF, Helmholtz Reciprocity

5. **Different surface types** - Lambertian, Hapke

6. **Shape from Shading (SfS)** - Hapke case, general case, reflectance maps, image irradiance equation

7. **Photometric Stereo** - least squares, multiple measurement variant, multiple light sources variant

8. **Computational molecules** - Sobel, Robert, Silver operators, finite differences (forward, backward, and average), Laplacians

9. **Lenses** - Thin lenses, thick lenses, telecentric lens, focal length, principal planes, pinhole model

10. **Patent Review** - Edge detector, Object Detection

1 Quick Mathematics Review

Let's quickly touch on some important mathematical theorems that are useful for machine vision.

- **Unconstrained optimization**: Suppose we are trying to optimize an objective $J(a, b, c)$ by some set of parameters $\{a, b, c\}$. We can find the best set of parameters $\{a^*, b^*, c^*\}$ using first-order conditions:

$$a^*, b^*, c^* = \arg\min_{a,b,c} J(a, b, c)$$
$$\rightarrow \frac{\partial J}{\partial a} = 0$$
$$\rightarrow \frac{\partial J}{\partial b} = 0$$
$$\rightarrow \frac{\partial J}{\partial c} = 0$$

  The solution a^*, b^*, c^* is the set of a, b, c that satisfies these first-order conditions. Note that this procedure is identical with maximization.

- **Green's Theorem** relates the line integral of a contour to the area located within that contour:

$$\oint_L (Ldx + Mdy) = \iint_D \left(\frac{\partial M}{\partial x} - \frac{\partial L}{\partial y}\right) dxdy$$

- **Bézout's Theorem**: *The maximum number of solutions is the product of the polynomial order of each equation in the system of equations*:

$$\text{number of solutions} = \prod_{e=1}^{E} o_e$$

- **Nyquist Sampling Theorem**: We must sample at twice the frequency of the highest-varying component of our image to avoid aliasing and consequently reducing spatial artifacts.

- **Taylor Series**: We can expand any analytical, continuous, infinitely-differentiable function into its Taylor Series form according to:

$$f(x + \delta x) = f(x) + \delta x f'(x) + \frac{(\delta x)^2}{2!} f''(x) + \frac{(\delta x)^3}{3!} f'''(x) + \frac{(\delta x)^4}{24} f^{(4)}(x) + ... = \sum_{i=0}^{\infty} \frac{(\delta x)^i f^{(i)}(x)}{i!}, \text{ where } 0! \triangleq 1$$

For a single variable function, and:

$$f(x + \delta_x, y + \delta_y) = f(x, y) + \delta_x \frac{\partial f(x, y)}{\partial x} + \delta_y \frac{\partial f(x, y)}{\partial y} + \cdots$$

For a multivariable function.

2 Projection: Perspective and Orthographic

A few important notes to remember:

- We use CAPITAL LETTERS for WORLD SPACE, and lowercase letters for image space.

- Orthographic projection is essentially the limit of perspective projection as we move the point we are mapping to the image plane far away from the optical axis.

- We can derive perspective projection from the pinhole model and similar triangles.

Perspective Projection:

$$\frac{x}{f} = \frac{X}{Z}, \frac{y}{f} = \frac{Y}{Z} \text{ (component form)}$$

$$\frac{1}{f}\mathbf{r} = \frac{1}{\mathbf{R} \cdot \hat{\mathbf{z}}}\mathbf{R} \text{ (vector form)}$$

Orthographic Projection:

$$x = \frac{f}{Z_0}X, y = \frac{f}{Z_0}y$$

3 Optical Flow

Optical flow captures the motion of an image over time in image space. Therefore, we will be interested in how brightness in the image changes with respect to (i) time, (ii) in the x-direction, and (iii) in the y-direction. We will use derivatives to describe these different quantities:

- $u \triangleq \frac{dx}{dt}$ - i.e. the velocity in the image space in the x direction.

- $v \triangleq \frac{dx}{dt}$ - i.e. the velocity in the image space in the y direction.

- $\frac{\partial E}{\partial x}$ - i.e. how the brightness changes in the x direction.

- $\frac{\partial E}{\partial y}$ - i.e. how the brightness changes in the y direction.

- $\frac{\partial E}{\partial t}$ - i.e. how the brightness changes w.r.t. time.

If $x(t)$ and $y(t)$ both depend on t, then the chain rule gives us the following for the total derivative:

$$\frac{dE(x,y,t)}{dt} = \frac{dx}{dt}\frac{\partial E}{\partial x} + \frac{dy}{dt}\frac{\partial E}{\partial y} + \frac{\partial E}{\partial t} = 0$$

Rewriting this in terms of u, v from above:

$$uE_x + vE_y + E_t = 0$$

This equation above is known as the **Brightness Change Constraint Equation (BCCE)**. This is also one of the most important equations in 2D optical flow.

Normalizing the equation on the right by the magnitude of the brightness derivative vectors, we can derive the **brightness gradient**:

$$(u,v) \cdot \left(\frac{E_x}{\sqrt{E_x^2 + E_y^2}}, \frac{E_y}{\sqrt{E_x^2 + E_y^2}} \right) = (u,v) \cdot \hat{\mathbf{G}} = -\frac{E_t}{\sqrt{E_x^2 + E_y^2}}$$

$$\textbf{Brightness Gradient}: \hat{\mathbf{G}} = \left(\frac{E_x}{\sqrt{E_x^2 + E_y^2}}, \frac{E_y}{\sqrt{E_x^2 + E_y^2}} \right)$$

What is the **brightness gradient**?

- A unit vector given by: $\left(\frac{E_x}{\sqrt{E_x^2 + E_y^2}}, \frac{E_y}{\sqrt{E_x^2 + E_y^2}} \right) \in \mathbb{R}^2$.

- Measures spatial changes in brightness in the image in the image plane x and y directions.

We are also interested in contours of constant brightness, or **isophotes**. These are curves on an illuminated surface that connects points of equal brightness (source: Wikipedia).

Finally, we are also interested in solving for optimal values of u and v for multiple measurements. In the ideal case without noise:

$$\begin{bmatrix} U \\ V \end{bmatrix} = \frac{1}{(E_{x_1}E_{y_2} - E_{y_1}E_{x_2})} \begin{bmatrix} E_{y_2} & -E_{y_1} \\ -E_{x_2} & E_{x_1} \end{bmatrix} \begin{bmatrix} -E_{t_1} \\ -E_{t_2} \end{bmatrix}$$

When there is noise we simply minimize our objective, instead of setting it equal to zero:

$$J(u,v) \triangleq \int_{x \in \mathbb{X}} \int_{y \in \mathbb{Y}} (uE_x + vE_y + E_t)^2 \, dx dy$$

We solve for our set of optimal parameters by finding the set of parameters that minimizes this objective:

$$u^*, v^* = \arg\min_{u,v} J(u,v) = \arg\min_{u,v} \int_{x \in \mathbb{X}} \int_{y \in \mathbb{Y}} (uE_x + vE_y + E_t)^2 \, dx dy$$

Since this is an unconstrained optimization problem, we can solve by finding the minimum of the two variables using two First-Order Conditions (FOCs):

- $\frac{\partial J(u,v)}{\partial u} = 0$

- $\frac{\partial J(u,v)}{\partial v} = 0$

Vanishing points: These are the points in the image plane (or extended out from the image plane) that parallel lines in the world converge to. Applications include:

- Multilateration

- Calibration Objects (Sphere, Cube)

- Camera Calibration

We will also look at **Time to Contact (TTC)**:

$$\frac{Z}{W} \triangleq= \frac{Z}{\frac{dZ}{dt}} = \frac{\text{meters}}{\frac{\text{meters}}{\text{seconds}}} = \text{seconds}$$

Let us express the inverse of this **Time to Contact (TTC)** quantity as C, which can be interpreted roughly as the number of frames until contact is made:

$$C \triangleq \frac{W}{Z} = \frac{1}{\text{TTC}}$$

4 Photometry

Here, we will mostly focus on some of the definitions we have encountered from lecture:

- **Photometry**: Photometry is the science of measuring visible radiation, light, in units that are weighted according to the sensitivity of the human eye. It is a quantitative science based on a statistical model of the human visual perception of light (eye sensitivity curve) under carefully controlled conditions.

- **Radiometry**: Radiometry is the science of measuring radiation energy in any portion of the electromagnetic spectrum. In practice, the term is usually limited to the measurement of ultraviolet (UV), visible (VIS), and infrared (IR) radiation using optical instruments.

- **Irradiance**: $E \triangleq \frac{\delta P}{\delta A}$ (W/m^2). This corresponds to light falling on a surface. When imaging an object, irradiance is converted to a grey level.

- **Intensity**: $I \triangleq \frac{\delta P}{\delta W}$ (W/ster). This quantity applied to a point source and is often directionally-dependent.

- **Radiance**: $L \triangleq \frac{\delta^2 P}{\delta A \delta \Omega}$ (W/m$^2 \times$ ster). This photometric quantity is a measure of how bright a surface appears in an image.

- **BRDF (Bidirectional Reflectance Distribution)**: $f(\theta_i, \theta_e, \phi_i, \phi_e) = \frac{\delta L(\theta_e, \phi_e)}{\delta E(\theta_i, \phi_i)}$. This function captures the fact that oftentimes, we are only interested in light hitting the camera, as opposed to the total amount of light emitted from an object. Last time, we had the following equation to relate image irradiance with object/surface radiance:

$$E = \frac{\pi}{4} L \left(\frac{d}{f}\right)^2 \cos^4 \alpha$$

  Where the irradiance of the image E is on the lefthand side and the radiance of the object/scene L is on the right. The BRDF must also satisfy **Helmholtz reciprocity**, otherwise we would be violating the 2nd Law of Thermodynamics. Mathematically, recall that Helmholtz reciprocity is given by:

$$f(\theta_i, \theta_e, \phi_i, \phi_e) = f(\theta_e, \theta_i, \phi_e, \phi_i) \; \forall \; \theta_i, \theta_e, \phi_i, \phi_e$$

5 Different Surface Types

With many of our photometric properties established, we can also discuss photometric properties of different types of surfaces. Before we dive in, it is important to also recall the definition of surface orientation:

- $p \triangleq \frac{\partial z}{\partial x}$

- $q \triangleq \frac{\partial z}{\partial y}$

- **Lambertian Surfaces**:

  - Ideal Lambertian surfaces are equally bright from all directions, i.e.

$$f(\theta_i, \theta_e, \phi_i, \phi_e) = f(\theta_e, \theta_i, \phi_e, \phi_i) \; \forall \; \theta_i, \theta_e, \phi_i, \phi_e$$
$$\text{AND}$$
$$f(\theta_i, \theta_e, \phi_i, \phi_e) = K \in \mathbb{R} \text{ with respect to } \theta_e, \phi_e$$

  - **"Lambert's Law"**:

$$E_i \propto \cos \theta_i = \hat{\mathbf{n}} \cdot \hat{\mathbf{s}}_i$$

- **Hapke Surfaces**:
  - The BRDF of a Hapke surface is given by:
  $$f(\theta_i, \phi_i, \theta_e, \phi_e) = \frac{1}{\sqrt{\cos \theta_e \cos \theta_i}}$$
  - Isophotes of Hapke surfaces are **linear** in spatial gradient, or (p, q) space.
- What do the isophotes of the moon look like, supposing the moon can fall under some of the different types of surfaces we have discussed?
  - **Lambertian**: We will see circles and ellipses of isophotes, depending on the angle made between the viewer and the the moon.
  - **Hapke**: Because of the BRDF behavior, isophotes will run **longitudinally** along the moon in the case in which it is a Hapke surface.

6 Shape from Shading (SfS)

This is one of the most fundamental problems in machine vision in which we try to estimate the surface of an object from brightness measurements. Several variations of this problem we consider:

- Hapke surfaces - this leads to linear isophotes in (p, q) space, and allows us to solve for one of the dimensions of interest.

- General case: There are several techniques we can apply to solve for this:
  - Green's Theorem (see lecture notes handout)
  - Iterative Discrete Optimization (see lecture notes handout)

  For these problems, we considered:
  - Characteristic strips (x, y, z, p, q)
  - Initial curves/base characteristics
  - Normalizing with respect to constant step sizes
  - A system of 5 ODEs
  - Stationary points and estimates of surfaces around them for initial points

Next, let us review reflectance maps. A **reflectance map** $R(p, q)$ is a lookup table (or, for simpler cases, a parametric function) that stores the brightness for particular surface orientations $p = \frac{\partial z}{\partial x}, q = \frac{\partial z}{\partial y}$.

The **Image Irradiance Equation** relates the reflectance map to the brightness function in the image $E(x, y)$ and is the first step in many Shape from Shading approaches.

$$E(x, y) = R(p, q)$$

7 Photometric Stereo

Photometric stereo is a technique in machine vision for estimating the surface normals of objects by observing that object under different lighting conditions. This problem is quite common in machine vision applications.

One way we can solve photometric stereo is by taking multiple brightness measurements from a light source that we move around. This problem becomes:

$$\begin{bmatrix} -s_1^T \\ -s_2^T \\ -s_3^T \end{bmatrix} \mathbf{n} = \begin{bmatrix} E_1 \\ E_2 \\ E_3 \end{bmatrix}$$

Written compactly:

$$\mathbf{Sn} = \mathbf{E} \longrightarrow \mathbf{n} = \mathbf{S}^{-1}\mathbf{E}$$

Note that we we need $\mathbf{S}$ to be invertible to compute this, which occurs when the light source vectors are not coplanar.

Other variations of this problem:

- Using light sources at different electromagnetic frequencies

- Using a camera with different light filters

- Moving the object to different locations

8   Computational Molecules

These are crucial components for applying machine concepts to real-life problems.

1. $E_x = \frac{1}{\epsilon}(E(x,y) - E(x-1,y))$ (Backward Difference), $\frac{1}{\epsilon}(z(x+1,y) - z(x,y))$ (Forward Difference)

2. $E_y = \frac{1}{\epsilon}(E(x,y) - E(x,y-1))$ (Backward Difference), $\frac{1}{\epsilon}(E(x,y+1) - E(x,y))$ (Forward Difference)

3. $\Delta E = \nabla^2 E = \frac{1}{\epsilon^2}(4E(x,y) - (E(x-1,y) + E(x+1,y) + E(x,y-1) + E(x,y+1)))$

4. $E_{xx} = \frac{\partial^2 E}{\partial x^2} = \frac{1}{\epsilon^2}(E(x-1,y) - 2(x,y) + E(x+1,y))$

5. $E_{yy} = \frac{\partial^2 z}{\partial y^2} = \frac{1}{\epsilon^2}(E(x,y-1) - 2(x,y) + E(x,y+1))$

6. **Robert's Cross**: This approximates derivatives in a coordinate system rotated 45 degrees (x', y'). The derivatives can be approximated using the $K_{x'}$ and $K_{y'}$ kernels:

$$\frac{\partial E}{\partial x'} \rightarrow K_{x'} = \begin{bmatrix} 0 & -1 \\ -1 & 0 \end{bmatrix}$$

$$\frac{\partial E}{\partial y'} \rightarrow K_{y'} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

7. **Sobel Operator**: This computational molecule requires more computation and it is not as high-resolution. It is also more robust to noise than the computational molecules used above:

$$\frac{\partial E}{\partial x} \rightarrow K_x = \begin{bmatrix} -1 & 0 & 1 \\ 2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

$$\frac{\partial E}{\partial y} \rightarrow K_y = \begin{bmatrix} -1 & 2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

8. **Silver Operators**: This computational molecule is designed for a hexagonal grid. Though these filters have some advantages, unfortunately, they are not compatible with most cameras as very few cameras have a hexagonal pixel structure.

9. **"Direct Edge" Laplacian**: $\frac{1}{\epsilon^2} \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$

10. **"Indirect Edge" Laplacian**: $\frac{1}{2\epsilon^2} \begin{bmatrix} 1 & 0 & 1 \\ 0 & -4 & 0 \\ 1 & 0 & 1 \end{bmatrix}$

11. **Rotationally-symmetric Laplacian**:

$$4\left(\frac{1}{\epsilon^2} \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}\right) + 1\left(\frac{1}{2\epsilon^2} \begin{bmatrix} 1 & 0 & 1 \\ 0 & -4 & 0 \\ 1 & 0 & 1 \end{bmatrix}\right) = \frac{1}{6\epsilon^2} \begin{bmatrix} 1 & 4 & 1 \\ 4 & -20 & 4 \\ 1 & 4 & 1 \end{bmatrix}$$

Some methods we used for analyzing how well these work:

1. **Taylor Series**: From previous lectures we saw that we could use averaging to reduce the error terms from 2nd order derivatives to third order derivatives. This is useful for analytically determining the error.

2. **Test functions**: We will touch more on these later, but these are helpful for testing your derivative estimates using analytical expressions, such as polynomial functions.

3. **Fourier domain**: This type of analysis is helpful for understanding how these "stencils"/molecules affect higher (spatial) frequency image content.
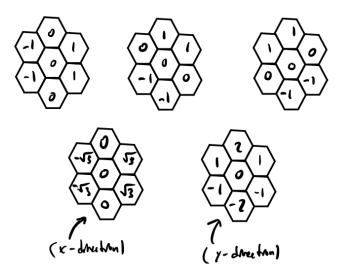
Figure 1: Silver Operators with a hexagonal grid.

9 Lenses

Lenses are also important, because they determine our ability to sense light and perform important machine vision applications. Some types of lenses:

- **Thin lenses** are the first type of lens we consider. These are often made from glass spheres, and obey the following three rules:
    - Central rays (rays that pass through the center of the lens) are undeflected - this allows us to preserve perspective projection as we had for pinhole cameras.
    - The ray from the focal center emerges parallel to the optical axis.
    - Any parallel rays go through the focal center.
- **Thick lenses** (cascaded thin lenses)
- **Telecentric lenses** - These "move" the the Center of Projection to infinity to achieve approximately orthographic projection.
- Potential distortions caused by lenses:
    - **Radial distortion**: In order to bring the entire angle into an image (e.g. for wide-angle lenses), we have the "squash" the edges of the solid angle, thus leading to distortion that is radially-dependent. Typically, other lens defects are mitigated at the cost of increased radial distortion. Some specific kinds of radial distortion [5]:
        * Barrel distortion
        * Mustache distortion
        * Pincushion distortion
    - **Lens Defects**: These occur frequently when manufacturing lenses, and can originate from a multitude of different issues.

10 Patent Review

The two patents we touched on were for:

- **Fast and Accurate Edge detection**: At a high level, this framework leveraged the following steps for fast edge detection:
    - CORDIC algorithm for estimating (and subsequently quantizing) the brightness gradient direction
    - Thresholding gradients via Non-Maximum Suppression (NMS)
    - Interpolation at the sub-pixel level along with peak finding for accurate edge detection at sub-pixel level
    - Bias compensation

    – Plane position

We also discussed some issues/considerations with the following aspects of this framework's design:

    – Quantization of gradient direction

    – Interpolation method for sub-pixel accuracy (quadratic, piecewise linear, cubic, etc.)

- **Object detection and pose estimation**: At a high level, this framework leveraged the following steps for fast object detection/pose estimation:

    – Finding specific **probing points** in a template image that we use for comparing/estimating gradients.

    – Using different **scoring functions** to compare template image to different configurations of the runtime image to find the object along with its pose (position with orientation).