6.801/6.866: Machine Vision, Lecture 10

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.

1 Lecture 10: Characteristic Strip Expansion, Shape from Shading, Iterative Solutions

In this lecture, we will continue studying our shape from shading problem, this time generalizing it beyond Hapke surfaces, making this framework amenable for a range of machine vision tasks. We will introduce a set of ordinary differential equations, and illustrate how we can construct a linear dynamical system that allows us to iteratively estimate the depth profile of an object using both position and surface orientation.

1.1 Review: Where We Are and Shape From Shading

Shape from Shading (SfS) is one of the class of problems we study in this course, and is part of a larger class of problems focusing on machine vision from image projection. In the previous lecture, we solved shape from shading for Hapke surfaces using the **Image Irradiance Equation** $E(x, y) = R(p, q)$, where $R(p, q)$ is the reflectance map that builds on the Bidirectional Reflectance Distribution Function (BRDF).

Recall additionally that irradiance (brightness in the image) depends on:

1. Illumination

2. Surface material of object being imaged

3. Geometry of the object being imaged

Recall from the previous lecture that for Hapke surfaces, we were able to solve the SfS problem in a particular direction, which we could then determine object surface orientation along. We can integrate along this direction (the profile direction) to find height z, but recall that we gain no information from the other direction.

For the Hapke example, we have a rotated coordinate system governed by the following ODEs (note that xi will be the variable we parameterize our profiles along):

1. X-direction: $\frac{dx}{d\xi} = p_s$

2. Y-direction: $\frac{dy}{d\xi} = q_s$

3. By chain rule, Z-direction: $\frac{dz}{d\xi} = \frac{\partial z}{\partial x}\frac{dx}{d\xi} + \frac{\partial z}{\partial y}\frac{dy}{d\xi} = pp_s + qq_s = p_s p + q_s q$

Intuition: Infinitesimal steps in the image given by ξ gives $\frac{dx}{d\xi}$, $\frac{dy}{d\xi}$, and we are interested in finding the change in height $\frac{dz}{d\xi}$, which can be used for recovering surface orientation.

Note: When dealing with brightness problems, e.g. SfS, we have implicitly shifted to orthographic projection ($x = \frac{f}{Z_0}X, y = \frac{f}{Z_0}y$). These methods can be applied to perspective projection as well, but the mathematics makes the intuition less clear. We can model orthographic projection by having a **telecentric lens**, which effectively places the object really far away from the image plane.

We can solve the 3 ODEs above using a **forward Euler** method with a given step size. For small steps, this approach will be accurate enough, and accuracy here is not too important anyway since we will have noise in our brightness measurements.

Recall brightness of Hapke surfaces is given by:

$$E = \sqrt{\frac{\hat{\mathbf{n}} \cdot \hat{\mathbf{s}}}{\hat{\mathbf{n}} \cdot \hat{\mathbf{v}}}} = \sqrt{\frac{\cos \theta_i}{\cos \theta_e}} =$$

$$frac{1 + p_s p + q_s q}{\sqrt{1 + p^2 + q^2}} r_s \cdot \frac{1}{\sqrt{1 + p_s p + q_s q}}$$

Some cancellation and rearranging yields:

$$p_s p + q_s q = r_s E^2 - 1$$

Therefore, we have a direct relationship between our measured brightness E and our unknowns of interest:

$$\frac{\partial z}{\partial x} p_s + \frac{\partial z}{\partial y} q_s = p p_s + q q_s = r_s E^2 - 1$$

Note here, however, that we do not know surface orientation based on this, since again for Hapke surfaces, we only know slope in one of our two directions in our rotated gradient space (p', q'). A forward Euler approach will generate a set of independent profiles, and we do not have any information about the surface orientation at the interfaces of these independent profiles. This necessitates an **initial curve** containing **initial conditions** for each of these independent profiles. In 3D, this initial curve is parameterized by η, and is given by: $(x(\eta), y(\eta), z(\eta))$. Our goal is to find $z(\eta, \xi)$, where:

- η parameterizes the length of the initial curve

- ξ parameterizes along the 1D profiles we carve out

Therefore, we are able to determine the slope in a particular direction.

1.2 General Case

Let us now generalize our Hapke case above to be any surface. We begin with our Image Irradiance Equation:

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

Let us now consider taking a small step $\delta x, \delta y$ in the image, and our goal is to determine how z changes from this small step. We can do so using differentials and surface orientation:

$$\delta z = \frac{\partial z}{\partial x} \delta x + \frac{\partial z}{\partial y} \delta y = p \delta x + q \delta y$$

Therefore, if we know p and q, we can compute δz for a given δx and δy. But in addition to updating z using the equation above, we will also need to update p and q (intuitively, since we are still moving, the surface orientation can change):

$$\delta p = \frac{\partial p}{\partial x} \delta x + \frac{\partial p}{\partial y} \delta y$$

$$\delta q = \frac{\partial q}{\partial x} \delta x + \frac{\partial q}{\partial y} \delta y$$

This notion of updating p and q provides motivation for keeping track/updating not only (x, y, z), but also p and q. We can think of solving our problem as constructing a **characteristic strip** of the ODEs above, composed of $(x, y, z, p, q) \in \mathbb{R}^5$. In vector-matrix form, our updates to p and q become:

$$\begin{bmatrix} \delta p \\ \delta q \end{bmatrix} = \begin{bmatrix} r & s \\ s & t \end{bmatrix} \begin{bmatrix} \delta x \\ \delta y \end{bmatrix} = \begin{bmatrix} \frac{\partial^2 z}{\partial x^2} & \frac{\partial^2 z}{\partial y \partial x} \\ \frac{\partial^2 z}{\partial x \partial y} & \frac{\partial^2 z}{\partial y^2} \end{bmatrix} \begin{bmatrix} \delta x \\ \delta y \end{bmatrix} = \begin{bmatrix} p_x & p_y \\ q_x & q_y \end{bmatrix} \begin{bmatrix} \delta x \\ \delta y \end{bmatrix} = \mathbf{H} \begin{bmatrix} \delta x \\ \delta y \end{bmatrix}$$

Where $\mathbf{H}$ is the Hessian of second spatial derivatives (x and y) of z. Note that from Fubuni's theorem and from our Taylor expansion from last lecture, $p_y = q_x$.

Intuition:

- First spatial derivatives $\frac{\partial z}{\partial x}$ and $\frac{\partial z}{\partial y}$ describe the **surface orientation** of an oject in the image.

- Second spatial derivatives $\frac{\partial^2 z}{\partial x^2}$, $\frac{\partial^2 z}{\partial y^2}$, and $\frac{\partial^2 z}{\partial x \partial y} = \frac{\partial^2 z}{\partial y \partial x}$ describe **curvature**, or the change in surface orientation.

- In 2D, the **curvature** of a surface is specified by the **radius of curvature**, or its inverse, **curvature.**

- In 3D, however, the curvature of a surface is specified by a Hessian matrix $\mathbf{H}$. This curvature matrix allows us to compute p and q.

One issue with using this Hessian approach to update p and q: how do we update the second derivatives r, s, and t? Can we use 3rd order derivatives? It turns out that seeking to update lower-order derivatives with higher-order derivatives will just generate increasingly more unknowns, so we will seek alternative routes. Let's try integrating our brightness measurements into what we already have so far:

$$\textbf{Image Irradiance Equation}: \quad E(x,y) = R(p,q)$$
$$\textbf{Chain Rule Gives}:$$
$$E_x = \frac{\partial E}{\partial x} = \frac{\partial R}{\partial p}\frac{\partial p}{\partial x} + \frac{\partial R}{\partial q}\frac{\partial q}{\partial x}$$
$$= \frac{\partial R}{\partial p}r + \frac{\partial R}{\partial q}s$$
$$E_y = \frac{\partial E}{\partial y} = \frac{\partial R}{\partial p}\frac{\partial p}{\partial y} + \frac{\partial R}{\partial q}\frac{\partial q}{\partial y}$$
$$= \frac{\partial R}{\partial p}s + \frac{\partial R}{\partial q}t$$

In matrix-vector form, we have:

$$\begin{bmatrix} E_x \\ E_y \end{bmatrix} = \begin{bmatrix} r & s \\ s & t \end{bmatrix} \begin{bmatrix} \frac{\partial R}{\partial p} \\[6pt] \frac{\partial R}{\partial q} \end{bmatrix} = \mathbf{H} \begin{bmatrix} \frac{\partial R}{\partial p} \\[6pt] \frac{\partial R}{\partial q} \end{bmatrix}$$

Notice that we have the same **Hessian** matrix that we had derived for our surface orientation update equation before!

Intuition: These equations make sense intuitively - brightness will be constant for constant surface orientation in a model where brightness depends only on surface orientation. Therefore, changes in brightness correspond to changes in surface orientation.

How do we solve for this Hessian $\mathbf{H}$? We can:

- Solve for E_x, E_y numerically from our brightness measurements.

- Solve for R_p, R_q using our reflectance map.

However, solving for $\mathbf{H}$ in this way presents us with 3 unknowns and only 2 equations (undetermined system). We cannot solve for $\mathbf{H}$, but we can still compute $\delta p, \delta q$. We can pattern match δx and δy by using a $(\delta x, \delta y)^T$ vector in the same direction as the gradient of the reflectance map, i.e.

$$\begin{bmatrix} \delta x \\ \delta y \end{bmatrix} = \begin{bmatrix} R_p \\ R_q \end{bmatrix} \xi$$

Where the vector on the lefthand side is our step in x and y, our vector on the righthand side is our gradient of the reflectance map in gradient space (p,q), and ξ is the step size. Intuitively, this is the direction where we can "make progress". Substituting this equality into our update equation for p and q, we have:

$$\begin{bmatrix} \delta p \\ \delta q \end{bmatrix} = \mathbf{H} \begin{bmatrix} R_p \\ R_q \end{bmatrix} \xi$$
$$= \begin{bmatrix} E_x \\ E_y \end{bmatrix} \delta \xi$$

Therefore, we can formally write out a system of 5 first-order ordinary differential equations (ODEs) that generate our characteristic strip as desired:

1. $\frac{dx}{d\xi} = R_p$

2. $\frac{dy}{d\xi} = R_q$

3. $\frac{dp}{d\xi} = E_x$

4. $\frac{dq}{d\xi} = E_y$

5. $\frac{dz}{d\xi} = pR_p + qR_q$ ("**Output Rule**")

Though we take partial derivatives on many of the righthand sides, we can think of these quantities as measurements or derived variations of our measurements, and therefore they do not correspond to partial derivatives that we actually need to solve for. Thus, this is why we claim this is a system of ODEs, and not PDEs.

A few notes about this system of ODEs:

- This system of 5 ODEs explores the surface along the characteristic strip generated by these equations.

- Algorithmically, we (a) Look at/compute the brightness gradient, which helps us (b) Compute p and q, which (c) Informs us of R_p and R_q for computing the change in height z.

- ODEs 1 & 2 and ODEs 3 & 4 are two systems of equations that each determine the update for the other system (i.e. there is dynamic behavior between these two systems). The 5th ODE is in turn updated from the results of updating the other 4 ODE quantities of interest.

- The step in the image $(\mathbf{x}, \mathbf{y})$ depends on the gradient of the reflectance map in $(\mathbf{p}, \mathbf{q})$.

- Analogously, the step in the reflectance map $(\mathbf{p}, \mathbf{q})$ depends on the gradient of the image in $(\mathbf{x}, \mathbf{y})$.

- This system of equations necessitates that we cannot simply optimize with block gradient ascent or descent, but rather a process in which we dynamically update our variables of interest using our other updated variables of interest.

- This approach holds generally for any reflectance map $R(p, q)$.

- We can express our image irradiance equation as a first order, *nonlinear* PDE:

$$E(x, y) = R\left(\frac{\partial z}{\partial x}, \frac{\partial z}{\partial y}\right)$$

Next, we will show how this general approach applied to a Hapke surface reduces to our previous SfS results for this problem.

1.2.1 Reducing General Form SfS for Hapke Surfaces

Recall reflectance map for Hapke surfaces:

$$R(p, q) = \sqrt{\frac{1 + p_s p + q_s q}{r_s}}$$

Taking derivatives using the system of 5 ODEs defined above, we have from the multivariate chain rule):

1. $\frac{\mathbf{dx}}{\mathbf{d\xi}} : R_p \triangleq \frac{\partial R}{\partial p} = \frac{1}{\sqrt{r_s}} \frac{1}{2\sqrt{1+p_s p+q_s q}} p_s$

2. $\frac{\mathbf{dy}}{\mathbf{d\xi}} : R_q \triangleq \frac{\partial R}{\partial q} = \frac{1}{\sqrt{r_s}} \frac{1}{2\sqrt{1+p_s p+q_s q}} q_s$

3. $\frac{\mathbf{dz}}{\mathbf{d\xi}} : pR_p + qR_q = \frac{p_s p+q_s q}{2\sqrt{r_s}\sqrt{1+p_s p+q_s q}}$

Since the denominator is common in all three of these derivative equations, we can just conceptualize this factor as a speed/step size factor. Therefore, we can simply omit this factor when we update these variables after taking a step. With this, our updates become:

1. $\delta x \leftarrow p_s$

2. $\delta y \leftarrow q_s$

3. $\delta z \leftarrow (p_s p + q_s q) = r_s E^2 - 1$

Which are consistent with our prior results using our Hapke surfaces. Next, we will apply this generalized approach to Scanning Electron Microscopes (SEMs):

1.2.2 Applying General Form SfS to SEMs

For SEMs, the reflectance map is spherically symmetric around the origin:

$$R(p, q) = f(p^2 + q^2) \text{ for some } f : \mathbb{R} \to \mathbb{R}$$

Using our system of 5 ODEs derived in the general case, we can compute our updates by first calculating derivatives $\frac{dx}{d\xi}$, $\frac{dy}{d\xi}$, and $\frac{dz}{d\xi}$:

1. $\frac{d\mathbf{x}}{d\xi} : R_p \triangleq \frac{\partial R}{\partial p} = 2f'(p^2 + q^2)p$

2. $\frac{d\mathbf{y}}{d\xi} : R_q \triangleq \frac{\partial R}{\partial q} = 2f'(p^2 + q^2)q$

3. $\frac{d\mathbf{z}}{d\xi} : pR_p + qR_q = 2f'(p^2 + q^2)(p^2 + q^2)$

Again, here we can also simplify these updates by noting that the term $2f'(p^2 + q^2)$ is common to all three derivatives, and therefore this factor can also be interpreted as a speed factor that only affects the step size. Our updates then become:

1. $\delta x \leftarrow p$

2. $\delta y \leftarrow q$

3. $\delta z \leftarrow p^2 + q^2$

This tells us we will be taking steps along the brightness gradient. Our solution generates **characteristic strips** that contain information about the surface orientation. To continue our analysis of this SfS problem, in addition to defining characteristic strips, it will also be necessary to define the base characteristic.

1.3 Base Characteristics

Recall that the characteristic strip that our system of ODEs solves for is given by a 5-dimensional surface composed of:

$$\begin{bmatrix} x & y & z & p & q \end{bmatrix}^T \in \mathbb{R}^5$$

Another important component when discussing our solution to this SfS problem is the **base characteristic**, which is the projection of the characteristic strip onto the x, y image plane:

$$\begin{bmatrix} x(\xi) & y(\xi) \end{bmatrix}^T = \text{projection}_{x,y}\{\textbf{characteristic strip}\} = \text{projection}_{x,y}\{\begin{bmatrix} x & y & z & p & q \end{bmatrix}^T\}$$

A few notes/questions about base characteristics?

- **What are base characteristics?** These can be thought of as the profiles in the (x, y) plane that originate (possibly in both directions) from a given point in the initial curve.

- **What are base characteristics used for?** These are used to help ensure we explore as much of the image as possible. We can interpolate/average (depending on how dense/sparse a given region of the image is) different base characteristics to "fill out" the solution over the entire image space.

- **How are base characteristics used in practice?** Like the solution profiles we have encountered before, base characteristics are independent of one another - this allows computation over them to be parallelizable. **Intuition**: These sets of base characteristics can be thought of like a **wavefront** propagating outward from the initial curve. We expect the solutions corresponding to these base characteristics to move at similar "speeds".

1.4 Analyzing "Speed"

When considering how the speeds of these different solution curves vary, let us consider ways in which we can set constant step sizes/speeds. Here are a few, to name:

1. **Constant step size in z**: This is effectively stepping "contour to contour" on a topographic map.
   Achieved by: Dividing by $pR_p + qR_q \implies \frac{dz}{d\xi} = 1$.

2. **Constant step size in image**: A few issues with this approach. First, curves may run at different rates. Second, methodology fails when $\sqrt{R_p^2 + R_q^2} = 0$.
   Achieved by: Dividing by $\sqrt{R_p^2 + R_q^2} \implies \frac{d}{d\xi}(\sqrt{\delta x^2 + \delta y^2}) = 1$.

3. **Constant Step Size in 3D/Object**: Runs into issues when $R_p = R_q = 0$.
Achieved by: Dividing by $\sqrt{R_p^2 + R_q^2 + (pR_p + qR_q)^2} \implies \sqrt{(\delta x)^2 + (\delta y)^2 + (\delta z)^2} = 1$.

4. **Constant Step Size in Isophotes**: Here, we are effectively taking constant steps in brightness. We will end up dividing by the dot product of the brightness gradient and the gradient of the reflectance map in (p, q) space.
Achieved by: Dividing by $(\frac{\partial E}{\partial x}\frac{\partial R}{\partial p} + \frac{\partial E}{\partial y}\frac{\partial R}{\partial q})\delta\xi = ((E_x, E_y) \cdot (R_p, R_q))\delta\xi \implies \delta E = 1$.

1.5 Generating an Initial Curve

While only having to measure a single initial curve for profiling is far better than having to measure an entire surface, it is still undesirable and is not a caveat we are satisfied with for this solution. Below, we will discuss how we can generate an initial curve to avoid having to measure one for our SfS problem.

As we discussed, it is undesirable to have to measure an initial curve/characteristic strip, a.k.a. measure:

$$\begin{bmatrix} x(\xi) & y(\xi) & z(\xi) & p(\xi) & q(\xi) \end{bmatrix}^T$$

But we do not actually need to measure all of these. Using the image irradiance equation we can calculate $\frac{\partial z}{\partial \eta}$:

$$E(x, y) = R(p, q)$$
$$\frac{\partial z}{\partial \eta} = \frac{\partial z}{\partial x}\frac{\partial x}{\partial \eta} + \frac{\partial z}{\partial y}\frac{\partial y}{\partial \eta}$$
$$= p\frac{\partial x}{\partial \eta} + q\frac{\partial y}{\partial \eta}$$

Where p and q in this case are our unknowns. Therefore, in practice, we can get by with just initial curve, and do not need the full initial characteristic strip - i.e. if we have $x(\eta), y(\eta)$, and $z(\eta)$, then we can compute the orientation from the reflectance map using our brightness measurements.

1.5.1 Using Edge Points to Autogenerate an Initial Curve

An even more sweeping question: do we even need an initial curve? Are there any special points on the object where we already know the orientation without a measurement? These points are along the **edge**, or **occluding boundary**, of our objects of interest. Here, we know the surface normal of each of these points. Could we use these edge points as "starting points" for our SfS solution?

It turns out, unfortunately, that we cannot. This is due to the fact that as we infinisimally approach the edge of the boundary, we have that $\frac{\partial z}{\partial x} \to \infty, \frac{\partial z}{\partial y} \to \infty$. Even though the slope $\frac{q}{p}$ is known, we cannot use it numerically/iteratively with our step updates above.

1.5.2 Using Stationary Points to Autogenerate an Initial Curve

Let us investigate another option for helping us to avoid having to generate an initial curve. Consider the brightest (or darkest) point on an image, i.e. the **unique, global, isolated extremum**. This point is a stationary point in (p, q) space, i.e.

$$\frac{\partial R}{\partial p} = \frac{\partial R}{\partial q} = 0, \text{ granted that } R(p, q) \text{ is differentiable.}$$

This would allow us to find the surface orientation (p, q) solely through optimizing the reflectance map function $R(p, q)$. One immediate problem with this stationary point approach is that we will not be able to step x and y, since, from our system of ODEs, we have that at our stationary points, our derivatives of x and y are zero:

$$\frac{dx}{d\xi} = R_p = 0$$
$$\frac{dy}{d\xi} = R_q = 0$$

Additionally, if we have stationary brightness points in image space, we encounter the "dual" of this problem. By definition stationary points in the image space imply that:

$$\frac{\partial E}{\partial x} = \frac{\partial E}{\partial y} = 0$$

6

This in turn implies that p and q cannot be stepped:

$$\frac{dp}{d\xi} = \frac{\partial E}{\partial x} = 0$$
$$\frac{dq}{d\xi} = \frac{\partial E}{\partial y} = 0$$

Intuition: Intuitively, what is happening with these two systems (each of 2 ODEs ((x, y) or p, q), as we saw in our 5 ODE system above) is that using stationary points in one domain amounts to updates in the other domain going to zero, which in turn prevents the other system's quantities of interest from stepping. Since δz depends on $\delta x, \delta y, \delta p, \delta q$, and since $\delta x, \delta y, \delta p, \delta q = 0$ when we use stationary points as starting points, this means we cannot ascertain any changes in δz along these points.

Note: The extremum corresponding to a stationary point can be **maximum**, as it is for Lambertian surfaces, or a **minimum**, as it is for Scanning Electron Microscopes (SEM).

However, even though we cannot use stationary points themselves as starting points, could we use the area around them? If we stay close enough to the stationary point, we can approximate that these neighboring points have nearly the same surface orientation.

- **Approach 1**: Construct a local tangent plane by extruding a circle in the plane centered at the stationary point with radius ϵ - this means all points in this plane will have the same surface orientation as the stationary point. Note that mathematically, a local 2D plane on a 3D surface is equivalent to a 2-manifold [1]. This is good in the sense that we know the surface orientation of all these points already, but not so great in that we have a degenerate system - since all points have the same surface orientation, under this model they will all have the same brightness as well. This prevents us from obtaining a unique solution.

- **Approach 2**: Rather than constructing a local planar surface, let us take a **curved surface** with **non-constant surface orientation** and therefore, under this model, **non-constant brightness**.

1.5.3   Example

Suppose we have a reflectance map and surface function given by:

$$\textbf{Reflectance Map}: \ R(p, q) = p^2 + q^2$$
$$\textbf{Surface Function}: \ z(x, y) = x^2 + y^2$$

Then, we can compute the derivatives of these as:

$$p = \frac{\partial z}{\partial x} = 2x$$
$$q = \frac{\partial z}{\partial y} = 4y$$
$$E(x, y) = R(p, q) = p^2 + q^2 = 4x^2 + 16y^2$$
$$E_x = \frac{\partial E}{\partial x} = 8x$$
$$E_y = \frac{\partial E}{\partial y} = 32y$$

The brightness gradient $(\frac{\partial E}{\partial x}, \frac{\partial E}{\partial y}) = (8x, 32y) = (0, 0)$ at the origin $x = 0, y = 0$.

Can we use the brightness gradient to estimate local shape? It turns out the answer is **no**, again because of stationary points. But if we look at the second derivatives of brightness:

$$E_{xx} = \frac{\partial^2 E}{\partial x^2} = \frac{\partial E}{\partial x}(8x) = 8$$
$$E_{yy} = \frac{\partial^2 E}{\partial y^2} = \frac{\partial E}{\partial y}(32y) = 32$$
$$E_{xy} = \frac{\partial^2 E}{\partial x \partial y} = \frac{\partial}{\partial x}(32y) = \frac{\partial}{\partial y}(8y) = 0$$

These second derivatives, as we will discuss more in the next lecture, will tell us some information about the object's shape. As we have seen in previous lectures, these second derivatives can be computed by applying computational molecules to our brightness measurements.

High-Level Algorithm

(NOTE: This will be covered in greater detail next lecture) Let's walk through the steps to ensure we can autogenerate an initial condition curve without the need to measure it:

1. Compute stationary points of brightness.

2. Use 2nd derivatives of brightness to estimate local shape.

3. Construct small cap (non-planar to avoid degeneracy) around the stationary point.

4. Begin solutions from these points on the edge of the cap surrounding the stationary point.

1.6   References

1. Manifold, https://en.wikipedia.org/wiki/Manifold

MIT OpenCourseWare
https://ocw.mit.edu

6.801 / 6.866 Machine Vision
Fall 2020

For information about citing these materials or our Terms of Use, visit: https://ocw.mit.edu/terms