Lecture 14: Curvature

Discrete Differential Geometry: An Applied Introduction

Keenan Crane • CMU 15-458/858
Curvature of Curves
Review: Curvature of a Plane Curve

• Informally, curvature describes “how much a curve bends”

• More formally, the curvature of an arc-length parameterized plane curve can be expressed as the rate of change in the tangent

\[ \kappa(s) := \langle N(s), \frac{d}{ds} T(s) \rangle \]

\[ = \langle N(s), \frac{d^2}{ds^2} \gamma(s) \rangle \]

Equivalently:

\[ \kappa(s) = \frac{d}{ds} \theta(s) \]

Here the angle brackets denote the usual dot product, i.e., \( \langle (a, b), (x, y) \rangle := ax + by \).
Review: Curvature and Torsion of a Space Curve

- For a plane curve, curvature captured the notion of "bending"
- For a space curve we also have torsion, which captures "twisting"

Intuition: torsion is "out of plane bending"
Fact. Up to rigid motions, an arc-length parameterized plane curve is uniquely determined by its curvature.

Q: Given only the curvature function, how can we recover the curve?

A: Just “invert” the two relationships $\frac{d}{ds} \theta = \kappa$, $\frac{d}{ds} \gamma = T$

First integrate curvature to get angle: $\theta(s) := \int_0^s \kappa(t) \, dt$

Then evaluate unit tangents: $T(s) := (\cos(\theta), \sin(\theta))$

Finally, integrate tangents to get curve: $\gamma(s) := \int_0^s T(t) \, dt$

Q: What about the rigid motion? Will this work for closed curves?
The fundamental theorem of space curves tells us we can also go the other way: given the curvature and torsion of an arc-length parameterized space curve, we can recover the curve itself.

In 2D we just had to integrate a single ODE; here we integrate a system of three ODEs—namely, Frenet-Serret!

\[
\frac{d}{ds} \begin{bmatrix} T \\ N \\ B \end{bmatrix} = \begin{bmatrix} 0 & -\kappa & 0 \\ \kappa & 0 & -\tau \\ 0 & \tau & 0 \end{bmatrix} \begin{bmatrix} T \\ N \\ B \end{bmatrix}
\]
Algorithm: Recover Plane Curve from Curvature

Fact. Up to rigid motions, a regular discrete plane curve is uniquely determined by its edge lengths and turning angles.

Q: Given only this data, how can we recover the curve?

A: Mimic the procedure from the smooth setting:

Sum curvatures to get angles: \( \varphi_{i,i+1} := \sum_{k=1}^{i} \theta_k \)

Evaluate unit tangents: \( T_{ij} := (\cos(\varphi_{ij}), \sin(\varphi_{ij})) \)

Sum tangents to get curve: \( \gamma_i := \sum_{k=1}^{i} \ell_{k,k+1} T_{k,k+1} \)

Q: Rigid motions?
Algorithm: Recover Space Curve from Curvature

TODO. Define discrete torsion, give algorithm.
Curvature of Surfaces
Weingarten Map

- The **Weingarten map** $dN$ is the differential of the Gauss map $N$
- At each point, tells us the change in the normal vector along any given direction $X$
- Since change in *unit* normal cannot have any component in the normal direction, $dN(X)$ is always tangent to the surface
- Can also think of it as a vector tangent to the unit sphere $S^2$

Q: Why is $dN(Y)$ “flipped”? 
Weingarten Map—Example

- Recall that for the sphere, $N = -f$. Hence, Weingarten map $dN$ is just $-df$:

$$f := (\cos(u) \sin(v), \sin(u) \sin(v), \cos(v))$$

$$df = \begin{pmatrix}
  -\sin(u) \sin(v), & \cos(u) \sin(v), & 0 \\
  \cos(u) \cos(v), & \cos(v) \sin(u), & -\sin(v)
\end{pmatrix} \, du +
\begin{pmatrix}
  \cos(u) \sin(v), & -\cos(u) \sin(v), & 0 \\
  -\cos(u) \cos(v), & -\cos(v) \sin(u), & \sin(v)
\end{pmatrix} \, dv$$

$$dN = \begin{pmatrix}
  \sin(u) \sin(v), & -\cos(u) \sin(v), & 0 \\
  -\cos(u) \cos(v), & -\cos(v) \sin(u), & \sin(v)
\end{pmatrix} \, du$$

Key idea: computing the Weingarten map is no different from computing the differential of a surface.
Normal Curvature

• For curves, curvature was the rate of change of the tangent; for immersed surfaces, we’ll instead consider how quickly the normal is changing.*

• In particular, normal curvature is rate at which normal is bending along a given tangent direction:

\[ \kappa_N(X) := \frac{\langle df(X), dN(X) \rangle}{|df(X)|^2} \]

• Equivalent to intersecting surface with normal-tangent plane and measuring the usual curvature of a plane curve

*For plane curves, what would happen if we instead considered change in $N$?
Normal Curvature—Example

Consider a parameterized cylinder:

\[ f(u, v) := (\cos(u), \sin(u), v) \]

\[ df = (-\sin(u), \cos(u), 0) \, du + (0, 0, 1) \, dv \]

\[ N = (-\sin(u), \cos(u), 0) \times (0, 0, 1) \]

\[ = (\cos(u), \sin(u), 0) \]

\[ dN = (-\sin(u), \cos(u), 0) \, du \]

\[ \kappa_N \left( \frac{\partial}{\partial u} \right) = \frac{\langle df \left( \frac{\partial}{\partial u} \right), dN \left( \frac{\partial}{\partial u} \right) \rangle}{|df \left( \frac{\partial}{\partial u} \right)|^2} = \frac{(-\sin(u), \cos(u), 0) \cdot (-\sin(u), \cos(u), 0)}{|(-\sin(u), \cos(u), 0)|^2} = 1 \]

\[ \kappa_N \left( \frac{\partial}{\partial v} \right) = \cdots = 0 \]

Q: Does this result make sense geometrically?
Principal Curvature

• Among all directions $X$, there are two principal directions $X_1$, $X_2$ where normal curvature has minimum/maximum value (respectively).

• Corresponding normal curvatures are the principal curvatures.

• Two critical facts*:

1. $g(X_1, X_2) = 0$

2. $dN(X_i) = \kappa_i df(X_i)$

Where do these relationships come from?
Shape Operator

- The change in the normal $N$ is always tangent to the surface
- Must therefore be some linear map $S$ from tangent vectors to tangent vectors, called the shape operator, such that
  \[
  df(SX) = dN(X)
  \]
- Principal directions are the eigenvectors of $S$
- Principal curvatures are eigenvalues of $S$
- Note: $S$ is not a symmetric matrix! Hence, eigenvectors are not orthogonal in $R^2$; only orthogonal with respect to induced metric $g$. 
Consider a nonstandard parameterization of the cylinder (*sheared* along z): 

\[ f(u,v) := (\cos(u), \sin(u), u + v) \]

\[ df = (-\sin(u), \cos(u), 1)du + (0, 0, 1)dv \]

\[ N = (\cos(u), \sin(u), 0) \]

\[ dN = (-\sin(u), \cos(u), 0)du \]

\[ df \circ S = dN \]

\[
\begin{bmatrix}
-\sin(u) & 0 \\
\cos(u) & 0 \\
1 & 1 \\
\end{bmatrix}
\begin{bmatrix}
S_{11} & S_{12} \\
S_{21} & S_{22} \\
\end{bmatrix}
= 
\begin{bmatrix}
-\sin(u) & 0 \\
\cos(u) & 0 \\
0 & 0 \\
\end{bmatrix}
\]

\[ S = \begin{bmatrix} 1 & 0 \\ -1 & 0 \end{bmatrix} \]

\[ X_1 = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \]

\[ X_2 = \begin{bmatrix} -1 \\ 1 \end{bmatrix} \]

\[ df(X_1) = (0, 0, 1) \]

\[ df(X_2) = (\sin(u), -\cos(u), 0) \]

\[ \kappa_1 = 0 \]

\[ \kappa_2 = 1 \]

**Key observation:** principal directions orthogonal only in \( \mathbb{R}^3 \).
Umbilic Points

- Points where principal curvatures are equal are called **umbilic points**
- Principal *directions* are not uniquely determined here
- What happens to the shape operator $S$?
  - May still have full rank!
  - Just have repeated eigenvalues, 2-dim. eigenspace

\[
S = \begin{bmatrix} 1/r & 0 \\ 0 & 1/r \end{bmatrix} \quad \kappa_1 = \kappa_2 = \frac{1}{r} \quad \forall X, SX = \frac{1}{r}X
\]

Could still of course choose (arbitrarily) an orthonormal pair $X_1, X_2$...
**Principal Curvature Nets**

- Walking along principal direction field yields **principal curvature lines**
- Collection of all such lines is called the **principal curvature network**
Separatrices and Spirals

• If we walk along a principal curvature line, where do we end up?

• Sometimes, a curvature line terminates at an umbilic point in both directions; these so-called **separatrices** (can) split network into regular patches.

• Other times, we make a closed loop. More often, however, behavior is *not* so nice!
Application—Quad Remeshing

• Recent approach to meshing: construct net *roughly* aligned with principal curvature—but with separatrices & loops, not spirals.

from Knöppel, Crane, Pinkall, Schröder, “Stripe Patterns on Surfaces”
Gaussian and Mean Curvature

Gaussian and mean curvature also fully describe local bending:

\[
\begin{align*}
\text{Gaussian} & : \quad K := \kappa_1 \kappa_2 \\
\text{mean*} & : \quad H := \frac{1}{2} (\kappa_1 + \kappa_2)
\end{align*}
\]

\( K > 0 \)  \quad \text{“developable”}  \quad \text{\( K = 0 \)}  \quad \text{\( K < 0 \)}

\( H \neq 0 \)  \quad \text{\( H \neq 0 \)}  \quad \text{\( H = 0 \)}  \quad \text{“minimal”}

*Warning: another common convention is to omit the factor of 1/2
Gaussian Curvature as Ratio of Ball Areas

- Originally defined Gaussian curvature as product of principal curvatures
- Can also view it as “failure” of balls to behave like Euclidean balls

Roughly speaking,

$$K \propto 1 - \frac{|B_g|}{|B_{\mathbb{R}^2}|}$$

More precisely:

$$|B_g(p, \varepsilon)| = |B_{\mathbb{R}^2}(p, \varepsilon)| \left(1 - \frac{K}{12} \varepsilon^2 + O(\varepsilon^3)\right)$$
Gauss-Bonnet Theorem

- Recall that the total curvature of a closed plane curve was always equal to $2\pi$ times the turning number $k$.

- **Q:** Can we make an analogous statement about surfaces?

- **A:** Yes! Gauss-Bonnet theorem says total Gaussian curvature is always $2\pi$ times the Euler characteristic $\chi$.

- For tori, Euler characteristic expressed in terms of the genus (number of "handles")
  \[ \chi := 2 - 2g \]

![Diagram of curves and surfaces with Euler characteristic values](image)
Total Mean Curvature?

**Theorem** (Minkowski): for a regular closed embedded surface,

\[ \int_M H \, dA \geq \sqrt{4\pi}A \]

**Q:** When do we get equality?

**A:** For a sphere.
Topological Invariance of Umbilic Count

Can classify regions around isolated umbilics into three types based on behavior of principal network: lemon, star, and monstar.

\[
\begin{align*}
\text{lemon} & \, (k_1) \\
\text{star} & \, (k_2) \\
\text{monstar} & \, (k_3)
\end{align*}
\]

**Fact.** If \(k_1, k_2, k_3\) are number of umbilics of each type, then

\[
k_1 - k_2 + k_3 = 2\chi
\]
Curvature of a Curve in a Surface

- Earlier, broke the “bending” of a space curve into curvature ($\kappa$) and torsion ($\tau$)
- For a curve in a surface, can instead break into normal and geodesic curvature:
  \[
  \kappa_n := \langle N_M, \frac{d}{ds} T \rangle \\
  \kappa_g := \langle B_M, \frac{d}{ds} T \rangle
  \]
- $T$ is still tangent of the curve; but unlike the Frenet frame, $N_M$ is the normal of the surface and $B_M := T \times N_M$

Q: Why no third curvature $\langle T_M, \frac{d}{ds} T \rangle$?
Second Fundamental Form

- Second fundamental form is closely related to principal curvature
- Can also be viewed as change in first fundamental form under motion in normal direction
- Why “fundamental?” First & second fundamental forms play role in important theorem...

\[ \mathbf{II}(X,Y) := \langle dN(X), df(Y) \rangle \]

\[ \kappa_N(X) := \frac{df(X), dN(X)}{|df(X)|^2} = \frac{\mathbf{II}(X,X)}{\mathbf{I}(X,X)} \]
Fundamental Theorem of Surfaces

• **Fact.** Two surfaces in $R^3$ are congruent if and only if they have the same first and second fundamental forms.

• …However, not every pair of bilinear forms $I, II$ on a domain $U$ describes a valid surface—must satisfy the **Gauss Codazzi equations**.

• Analogous to fundamental theorem of plane curves: determined up to rigid motion by curvature.

• …However, for *closed* curves not every curvature function is valid (*e.g.*, must integrate to $2k\pi$).
Fundamental Theorem of Discrete Surfaces

- **Fact.** Up to rigid motions, can recover a discrete surface from its *dihedral angles* and *edge lengths*.

- Fairly natural analogue of Gauss-Codazzi; data is split into edge lengths (encoding $\mathbf{I}$) and dihedral angles (encoding $\mathbf{II}$)

- Basic idea: construct each triangle from edge lengths; use dihedral angles to globally glue together

from Wang, Liu, and Tong, “Linear Surface Reconstruction from Discrete Fundamental Forms on Triangle Meshes”
Other Descriptions of Surfaces?

• Classic question in differential geometry:

“What data is sufficient to completely determine a surface in space?”

• Many possibilities…
  • First & second fundamental form (Gauss-Codazzi)
  • Mean curvature and metric (up to “Bonnet pairs”)
  • Convex surfaces: metric alone is enough (Alexandrov/Pogorolev)
  • Gauss curvature essentially determines metric (Kazdan-Warner)
• …in general, still a surprisingly murky question!
Open Challenges in Shape Recovery

• What other **discrete** quantities determine a surface?

• …and how can we (efficiently) recover a surface from this data?

• Lengths + dihedral angles work in general (*fundamental theorem of discrete surfaces*); lengths alone are sufficient for convex surfaces. What about just dihedral angles?

• Have a variety of discrete curvatures. Which are sufficient, for which classes of surfaces?

• Why bother? Offers new & different ways to analyze, process, edit, transmit, … curved surfaces digitally.

from Eigensatz & Pauly, “Curvature Domain Shape Processing”
Thanks!

**Discrete Differential Geometry:**

**An Applied Introduction**

Keenan Crane • CMU 15-458/858