CSE 274
Discrete Differential Geometry
(Selected Topics in Computer Graphics)

Instructor: Albert Chern
TA: Mohammad Nezhad
Discrete Differential Geometry

• **Goal:** Differential geometric notions and their discrete theories for geometry processing and modeling.

• **Prerequisite:** Linear algebra, Multivariable calculus, (computer graphics).

• **Grade:** 4 homework assignments (theory+implementation) (90%) and participation (10%).

• **Collaboration:** Team of one (individual) or two people.
CSE 274 - Selected Topics in Graphics - Chern [FA20]


Zoom Classrooms:

The lectures are held on Mon, Wed, Fri 17:00-17:50. The first lecture will be on Oct 2 (Fri) and the last lecture will be on Dec 11 (Fri).

Lecture Zoom link: Go to Zoom LTI PRO in the Canvas menu.
Office hour (Instructor Albert Chern): TBD
Office hour (TA Mohammad Nezhad): TBD

The lecture will be recorded and be available on Canvas (Zoom LTI PRO, Cloud Recording). It won't be available in public. This is an option for you to replay in case a smooth live streaming is not available to you.

Getting Started
Course Information

- **Piazza** (Q&A discussion)
  
  - [https://piazza.com/signup/eng.ucsd](https://piazza.com/signup/eng.ucsd)
Course Information


CSE 274 (FA 2020)


The course provides an introduction to discrete differential geometry and its applications in geometric modeling and analysis. The contents include the smooth and discrete theory of curves, surfaces, exterior calculus, the Hodge theory, and the vector bundle theory. The theories are explained alongside with applications including numerical methods for differential equations on manifolds, surface texturing, shape analysis and vector field designs. The course also covers the basics of the graphics software Houdini FX.

Course Logistics

The main course information, Zoom links are on the [UCSD Canvas](https://canvas.ucsd.edu).  

- **Instructor:** Albert Chern (office hour: TBD)  
  - **TA:** Mohammad Nezhad (office hour: TBD)  
  - **Time:** MWF 17:00~17:50 (remote)  
  - **Grade:** 90% Homework (including theory and implementation) + 10% Participation  
    - We recommend the implementation part of the homework be done via the Houdini software.  
  - Use [Piazza](https://piazza.com) for Q&A discussion.  
  - Use [Gradescope](https://gradescope.com) linked from Canvas for homework submission.

Schedule

- **10/2 (Fri):** Introduction to Discrete Differential Geometry.  
- **10/5 (Mon):** Intro to Houdini. (HW0 available)  
- **10/7 (Wed):** About quaternions.  
- **10/9 (Fri):** Plane curves.  
- **10/12 (Mon):** Discrete plane curves. (HW0 due)  
- **10/14 (Wed):** Space curves. (HW1 available)  
- **10/16 (Fri):** Space curves.  
- **10/19 (Mon):** Discrete surface theory.
Course Information

- **Office hour**: we will make a poll (until 10/4 Sun) on Piazza for the time slot.
Course Information

- **Lecture note:** [http://cseweb.ucsd.edu/~alchern/teaching/DDG.pdf](http://cseweb.ucsd.edu/~alchern/teaching/DDG.pdf)
Course Information

- Implementation part of the homework: Houdini Software
Course Information

- Implementation part of the homework: **Houdini Software**
  - Tutorial next Monday
  - HW0 due next-next Monday: upload a screenshot/render of your surface.
• Install Houdini & SciPy (numerical linear algebra for sparse matrices)

Houdini FX
Houdini is a 3D animation software by SideFX that is broadly used in filming industry. Its built-in shader language (VEX), geometric data structures, and its integration with Python (where one can call numerical linear algebra library such as SciPy) make it a powerful tool for scientific computing and geometry processing. Moreover Houdini comes with industry standard renderer allowing stunning visualization. Its free apprentice license is fully functional, with only a few limitations in the resolution of rendering preventing commercial uses. Houdini has become a frequently used software for not only research demonstration, but also for teaching and generating illustrations for lectures and talks.

Lecture Note
• Houdini Introduction Lecture Note
• Houdini Tech Blog

Getting Started
To get started, first we recommend rummaging around the SideFX website and searching for tutorials on YouTube.

We also recommend that you work through some of the tutorials on the Houdini Tech Blog, a course blog for "Visualization in Mathematics" in TU Berlin. For example you can work through all the tutorials in the "Introduction" category. There are more tutorials on that site and if you feel inspired check them out. All come with Houdini files that allows you to load the completed tutorials if you wish.

Our lecture note listed above teaches you how to create simple geometry inside Houdini using the VEX language; basic scene setup with simple lighting and camera as well as manipulating surface appearance for rendering.

The complete documentation is also available on the SideFX website.

A note on installing Houdini (and SciPy)

1. Find the Python directory. Open Houdini. On any dock (subwindow) add a tab "Python Shell." In the Python shell, type

```python
>>> import os
>>> pathos = path.abspath(os._file_
>>> print(path)
```

The result should be something like

/Applications/Houdini/18.0.499/Frameworks/Python.framework/Versions/Current/Lib/python2.7/os.py

2. Open Terminal, and goto the bin directory ".../Frameworks/Python.framework/Versions/Current/bin" (according to the path shown on Step 1.)

```
$ cd /Applications/Houdini/18.0.499/Frameworks/Python.framework/Versions/Current/bin
```

3. Use the local python to install the Python Package Installer (pip). In that directory execute (you will need to enter your root password for the superuser do (sudo) command)

```
$ sudo curl https://bootstrap.pypa.io/get-pip.py -o get-pip.py
$ sudo ./python get-pip.py
```

4. Use the local pip to install SciPy.

```
$ sudo ./pip install scipy
```

5. Back to the Houdini Python shell (no need to restart Houdini). Now

```python
>>> import scipy
```

should give no error.
Course Information

• Install Houdini & SciPy (numerical linear algebra for sparse matrices)
Questions?
What is Geometry?

- Relationships between shapes, order of spaces
- Symmetries and invariants
What is Geometry?

• Relationships between shapes, order of spaces
What is Geometry?

- Relationships between shapes, order of spaces (18th, 19th century)
- Engineering design
- General relativity
- Computer graphics

Differential geometry

Gauss, Riemann, Poincaré, …

Schwarz 1890

photo courtesy Capanna
What is Geometry?

- Relationships between shapes, order of spaces (18th, 19th century)

Non-Euclidean projective geometry

- Two points connect into a line
- Center and distance gives a circle
- 90° is well-defined
- Two lines meet unless parallel

Lobachevsky, Möbius, Grassmann, Beltrami, Lie, Plücker, Cayley, Klein,…
What is Geometry?

- Relationships between shapes, order of spaces (18th, 19th century)

Non-Euclidean projective geometry

Lobachevsky, Möbius, Grassmann, Beltrami, Lie, Plücker, Cayley, Klein,…

- Linear algebra
- Quantum mechanics
- Algebraic geometry
What is Geometry?

- Relationships between shapes, order of spaces
- Symmetries and invariants

**Klein’s Erlangen program 1872:**

Ingredients of geometry:

- Collection of objects
- Group of transformations
- Invariant notions
What is Geometry?

Ingredients of geometry:
- Collection of objects
- Group of transformations
- Invariant notions
  - straight lines, planes
  - distance
  - circles, spheres
  - angles
  - area
  - parallelism
- Non-invariant notions
  - additions $\mathbf{p}_1 + \mathbf{p}_2$
  - dot product $\mathbf{p}_1 \cdot \mathbf{p}_2$

Geometry in Euclidean space
What is Geometry?

Ingredients of geometry:
- Collection of objects
- Group of transformations

\[
\begin{align*}
\mathbf{p} &= \begin{bmatrix} x \\ y \\ z \end{bmatrix} \in \mathbb{R}^3 \\
\tilde{\mathbf{p}} &= R\mathbf{p} \\
\text{rotation}
\end{align*}
\]

- Invariant notions
  - straight lines, planes
  - distance
  - circles, spheres
  - angles
  - area
  - parallelism
  - additions $\mathbf{p}_1 + \mathbf{p}_2$
  - dot product $\mathbf{p}_1 \cdot \mathbf{p}_2$

- Non-invariant notions

Geometry in Euclidean vector space
What is Geometry?

**Ingredients of geometry:**
- Collection of objects
- Group of transformations

\[
\left\{ \begin{array}{c}
  p = \begin{bmatrix}
    x \\
    y \\
    z
  \end{bmatrix} \\
  \in \mathbb{R}^3
\end{array} \right\}
\]

\[\tilde{p} = Ap\]

invertible matrix

**Invariant notions**
- straight lines, planes
- parallelism
- additions \( p_1 + p_2 \)

**Non-invariant notions**
- distance
- circles, spheres
- angles
- area
- dot product \( p_1 \cdot p_2 \)

**Geometry in vector space**
What is Geometry?

Ingredients of geometry:

- Collection of objects
- Group of transformations

\[
\begin{align*}
\mathbf{p} &= \begin{bmatrix} x \\ y \\ z \end{bmatrix} \in \mathbb{R}^3 \\
\tilde{\mathbf{p}} &= A\mathbf{p} + \mathbf{b}
\end{align*}
\]

- Invariant notions
  - straight lines, planes
  - parallelism
  - midpoint \( \frac{\mathbf{p}_1 + \mathbf{p}_2}{2} \)

- Non-invariant notions
  - distance
  - circles, spheres
  - angles
  - area
  - additions \( \mathbf{p}_1 + \mathbf{p}_2 \)
  - dot product \( \mathbf{p}_1 \cdot \mathbf{p}_2 \)

Geometry in affine space
What is Geometry?

Ingredients of geometry:
- Collection of objects
  - oriented circles in $\mathbb{R}^2$
- Group of transformations
  - rotate, translate
  - normal offset (propagate wave)
- Invariant notions
  - straight lines
  - oriented contact
  - lengths of common tangents
- Non-invariant notions
  - intersection
  - point

Laguerre geometry of oriented circles
What is Geometry?

Ingredients of geometry:

- Collection of objects
- Group of transformations
- Invariant notions
  - tangents, normals
  - curvatures
- Non-invariant notions
  - particle speed
  - particle acceleration

Differential geometry of curves
Differential Geometry

Why do we care?

- geometry of surfaces

Konaković, Crane, Deng, Bouaziz, Piker & Pauly 2016

Jiang, Wang, Rist, Wallner & Pottmann 2020

Sageman-Furnas, Chern, Ben-Chen, Vaxman 2019
Differential Geometry

Why do we care?

- geometry of surfaces
- mothertongue of physical theories

Elasticity of lipid bilayer membrane (Helfrich 1973)

\[ E = \int_{S} \alpha \, dA + \beta (H - H_0)^2 \, dA + \gamma K \, dA \]

- mean curvature
- Gaussian curvature
Differential Geometry

Why do we care?

- geometry of surfaces
- mothertongue of physical theories
- computation: simulation, processing

Chern, Knöppel, Pinkall & Schröder 2017

Stein, Grinspun, Wardetzky & Jacobson 2018

Bergou, Wardetzky, Robinson, Audoly & Grinspun 2008

Solomon, de Goes, Peyré, Cuturi, Butscher, Nguyen, Du & Guibas 2015
Differential Geometry

How do we apply differential geometry ideas?

• Surfaces are collections of samples and connectivity
• Apply continuous ideas
  BUT: setting is discrete
• What is the right way?
  ▶ Discrete v.s. Discretized
Discretized v.s. Discrete

modeling & analysis
Invariants, Symmetries

discretization
Approximation theory

discrete
differential
geometry
Discretized

Build smooth manifold structure

- Collection of charts with compatibility on their overlaps
- Realize everything as smooth functions
  - Polynomial, rational functions
  - Spectral basis
- Numerical analysis paradigm
  - Truncation error
  - Stability
Example: pendulum equation.

\[ m\ddot{\theta} = -mg \sin \theta \]

✓ Energy conservation

\[ \frac{1}{2} m\dot{\theta}^2 - mg \cos \theta = \text{const} \]

✓ Integrable system
Discretized

Example: pendulum equation.

\[ m\ddot{\theta} = -mg \sin \theta \]

- High order differential equation solver
  (4th order Runge–Kutta method)

Given \((\theta_i, v_i = \dot{\theta}_i)\)

\[
\begin{align*}
\theta_{i+1/2}^* &= \theta_i + \frac{\Delta t}{2} v_i \\
\dot{v}_{i+1/2}^* &= v_i - \frac{\Delta t}{2} \sin \theta_i \\
\theta_{i+1/2}^{**} &= \theta_i + \frac{\Delta t}{2} \dot{v}_{i+1/2}^* \\
\dot{v}_{i+1/2}^{**} &= \dot{v}_{i+1/2}^* - \frac{\Delta t}{2} \sin \theta_{i+1/2}^* \\
\theta_{i+1/2}^{***} &= \theta_i + \Delta t \dot{v}_{i+1/2}^{**} \\
\dot{v}_{i+1/2}^{***} &= \dot{v}_{i+1/2}^{**} - \Delta t \sin \theta_{i+1/2}^{**}
\end{align*}
\]

Output

\[
\begin{align*}
\theta_{i+1} &= \theta_i + \frac{\Delta t}{6} \left( v_i + 2\dot{v}_{i+1/2}^* + 2\dot{v}_{i+1/2}^{**} + \dot{v}_{i+1/2}^{***} \right) \\
v_{i+1} &= v_i - \frac{\Delta t}{6} \left( \sin \theta_i + 2\sin \theta_{i+1/2}^* + 2\sin \theta_{i+1/2}^{**} + \sin \theta_{i+1/2}^{***} \right)
\end{align*}
\]
Discretized

Example: pendulum equation.

\[ m\ddot{\theta} = -mg \sin \theta \]

- High order differential equation solver (4th order Runge–Kutta method)

Given \((\theta_i, v_i = \dot{\theta}_i)\)

\[ \begin{align*}
\theta_{i+1/2}^* &= \theta_i + \frac{\Delta t}{2} v_i \\
v_{i+1/2}^* &= v_i - \frac{\Delta t}{2} \sin \theta_i
\end{align*} \]

\[ \begin{align*}
\theta_{i+1/2}^{**} &= \theta_i + \frac{\Delta t}{2} v_{i+1/2} \\
v_{i+1/2}^{**} &= v_i - \frac{\Delta t}{2} \sin \theta_{i+1/2}^*
\end{align*} \]

\[ \begin{align*}
\theta_{i+1}^{***} &= \theta_i + \Delta t v_{i+1/2}^{**} \\
v_{i+1}^{***} &= v_i - \Delta t \sin \theta_{i+1/2}^{**}
\end{align*} \]

Output \((\theta_{i+1}, v_{i+1} = \dot{\theta}_{i+1})\)

\[ \begin{align*}
\theta_{i+1} &= \theta_i + \frac{\Delta t}{6} \left( v_i + 2v_{i+1/2}^* + 2v_{i+1/2}^{**} + v_{i+1}^{***} \right) \\
v_{i+1} &= v_i - \frac{\Delta t}{6} \left( \sin \theta_i + 2 \sin \theta_{i+1/2}^* + 2 \sin \theta_{i+1/2}^{**} + \sin \theta_{i+1}^{***} \right)
\end{align*} \]

\(\Delta t = 0.1\)
Discretized

Example: pendulum equation.

\[ m\ddot{\theta} = -mg \sin \theta \]

- High order differential equation solver
  (4th order Runge–Kutta method)

Given \((\theta_i, v_i = \dot{\theta}_i)\)

\[ \begin{align*}
\theta_{i+1/2}^* &= \theta_i + \frac{\Delta t}{2} v_i \\
v_{i+1/2}^* &= v_i - \frac{\Delta t}{2} \sin \theta_i \\
\theta_{i+1/2}^{**} &= \theta_i + \frac{\Delta t}{2} v_{i+1/2}^* \\
v_{i+1/2}^{**} &= v_i - \frac{\Delta t}{2} \sin \theta_{i+1/2}^* \\
\theta_{i+1}^{***} &= \theta_i + \Delta t v_{i+1/2}^{**} \\
v_{i+1}^{***} &= v_i - \Delta t \sin \theta_{i+1/2}^{**}
\end{align*} \]

Output

\[ \begin{align*}
\theta_{i+1} &= \theta_i + \frac{\Delta t}{6} \left( v_i + 2v_{i+1/2}^* + 2v_{i+1/2}^{**} + v_{i+1}^{***} \right) \\
v_{i+1} &= v_i - \frac{\Delta t}{6} \left( \sin \theta_i + 2 \sin \theta_{i+1/2}^* + 2 \sin \theta_{i+1/2}^{**} + \sin \theta_{i+1}^{***} \right)
\end{align*} \]

\( \Delta t = 0.9 \)

Structure not preserved
Example: pendulum equation.

\[ m\ddot{\theta} = -mg \sin \theta \]

- A 2nd order discretization

\[ \frac{\theta_{i-1} - 2\theta_i + \theta_{i+1}}{\Delta t^2} = -\sin \theta_i \]
Example: pendulum equation.

\[ m\ddot{\theta} = -mg \sin \theta \]

- A 2nd order discretization

\[ \frac{\theta_{i-1} - 2\theta_i + \theta_{i+1}}{\Delta t^2} = -\sin \theta_i \]

✔ Energy conservation in “asteroid belts”
Example: pendulum equation.

\[ m\ddot{\theta} = -mg \sin \theta \]

- A 2nd order discretization

\[
\frac{\theta_{i-1} - 2\theta_i + \theta_{i+1}}{\Delta t^2} = -\sin \theta_i
\]

Least action principle

\[
\int \left( \frac{m}{2} \dot{\theta}^2 + mg \cos \theta \right) \, dt
\]

First introduce discrete action, then derive the least action paths

✓ Energy conservation in “asteroid belts”

\[ \Delta t = 0.9 \]
Discrete

Example: pendulum equation.

\[ m\ddot{\theta} = -mg \sin \theta \]

- Another 2nd order discretization

\[ \frac{\theta_{i-1} - 2\theta_i + \theta_{i+1}}{\Delta t^2} = -\sin \theta_i \]

\[ \frac{\theta_{i-1} - 2\theta_i + \theta_{i+1}}{\Delta t^2} = 4 \text{arg} \left( 1 + \frac{\Delta t^2}{4} e^{-i\theta_i} \right) \]

✓ Integrable system
Example: pendulum equation.
\[ m\ddot{\theta} = -mg \sin \theta \]

- Another 2nd order discretization
\[ \frac{\theta_{i-1} - 2\theta_i + \theta_{i+1}}{\Delta t^2} = -\sin \theta_i \]
\[ \frac{\theta_{i-1} - 2\theta_i + \theta_{i+1}}{\Delta t^2} = 4 \arg \left( 1 + \frac{\Delta t^2}{4} e^{-i\theta_i} \right) \]

✓ Integrable system
Discretized v.s. Discrete

- General
  - No-structure
  - Quantitative high-precision

4th order Runge-Kutta

Variational integrator

Discrete integrable system

- Rare
  - Structure preserving
  - Qualitative exact

$\Delta t = 0.1$

$\Delta t = 0.9$
Discretized v.s. Discrete

Invariants, Symmetries

Approximation theory

Structure preserving: There are properties which are exact rather than approximation.
Discretized v.s. Discrete

Pellis & Pottmann 2018

Wang, Pellis, Rist, Pottmann & Müller 2019

Yas Marina hotel in Abu Dhabi

Smithsonian American Art Museum
Discrete Differential Geometry

Building from the ground up

• Discrete geometry is given
  ▶ Triangle, tetrahedral mesh

• How to do calculus?
  ▶ Preserve crucial properties, e.g. \( \int_{a}^{b} f'(x) \, dx = f(b) - f(a) \)
Discrete Differential Geometry

Building from the top down

- High level theorems
  - Gauss Theorema Egregium
  - Riemann mapping theorem
  - Least action principle

\[ f : \Omega_1 \subset \mathbb{C} \rightarrow \Omega_2 \subset \mathbb{C} \]
Things we will cover

• Curves and Surfaces
• Exterior Calculus
• Hodge Decomposition
• Vector Field Design
Curves and Surfaces

curvature
torsion
theorems of Gauss
Exterior Calculus
coordinate-free calculus
scalar, flux, density
PDEs
Laplacian
Curvature Flow
Hodge decomposition
linear subspaces of differential forms
linear algebraic topology
Vector Field Design
vectors: amplitude & phase
gauge theory
quantum mechanics