What's next in your graphics journey?

Logistics

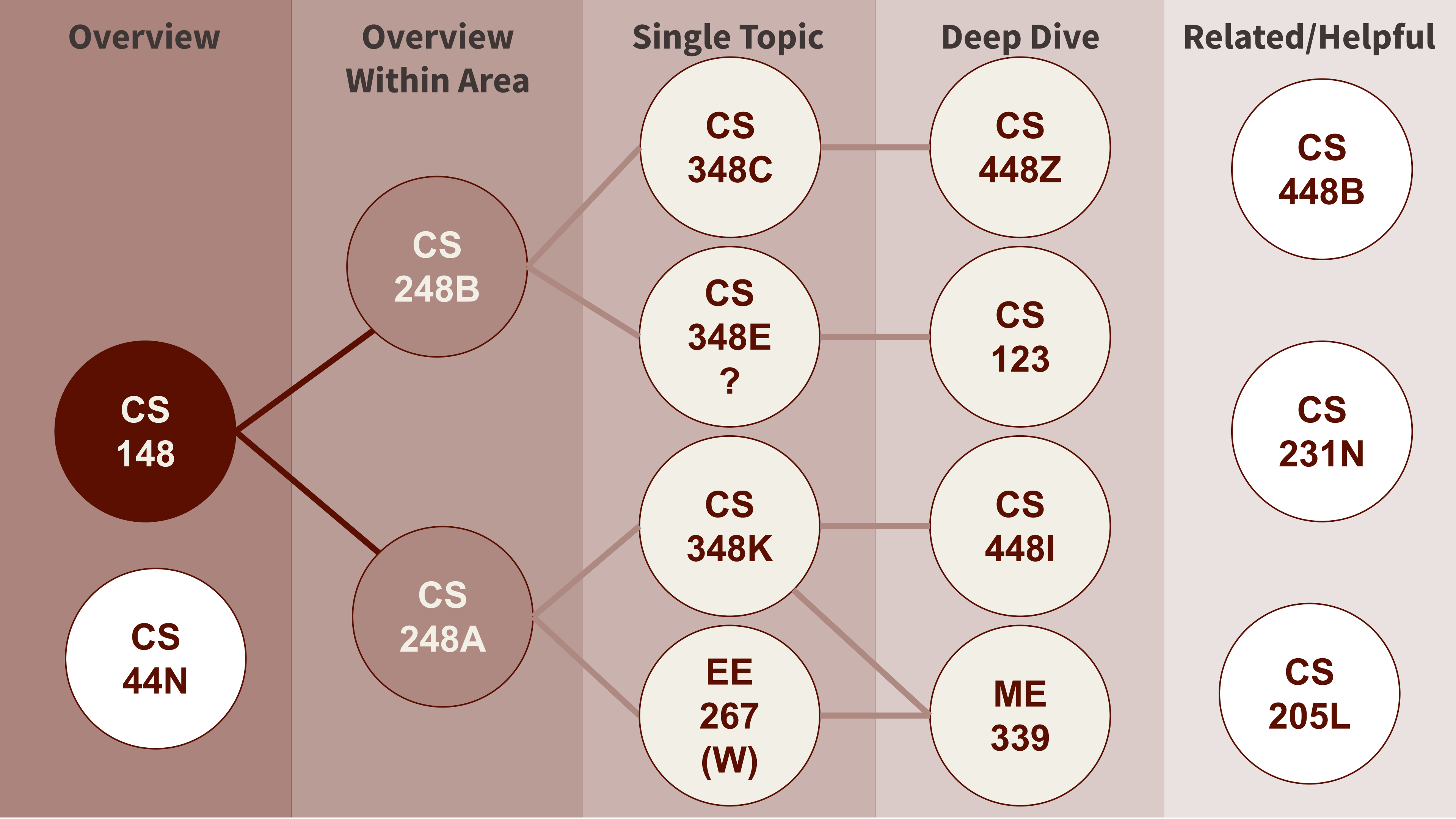
- Course Reviews
  - in Canvas, students will see a pop-up notification on their Canvas dashboard page any time they log into Canvas during the evaluation period.
  - in Axess > My Academics > Course and Section Evaluations > Link to the evaluation system near the top of the page
- Final Project
  - CHECK THE ED FORUM IT HAS EVERYTHING!!!!!
  - https://edstem.org/us/courses/99769/discussion/8192013
- Final Project Rendering
  - Render now !!!

Lecture Outline

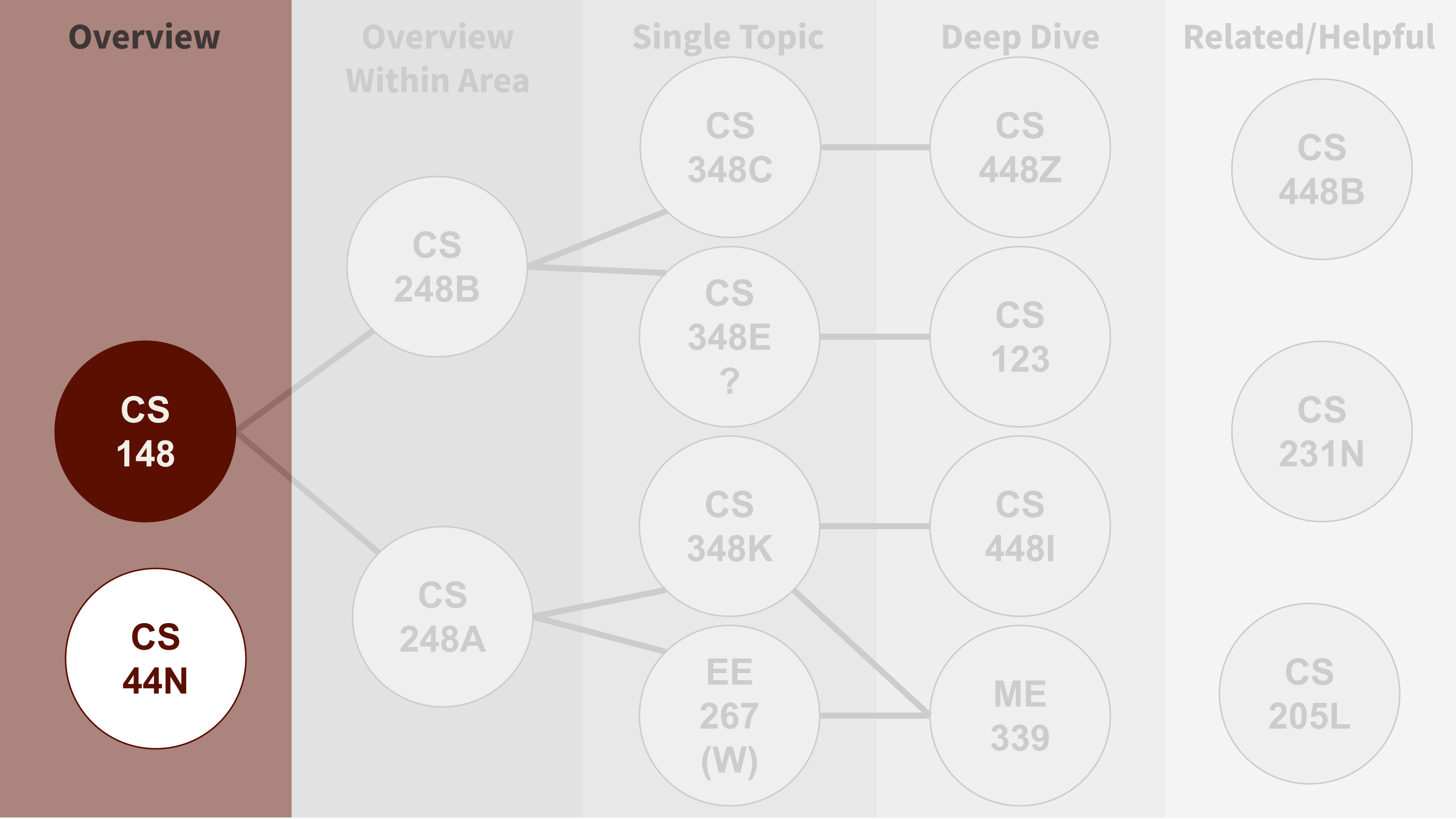
- Classes
- Cutting Edge Graphics (SIGGRAPH)
- Ray Tracing Optimization
- Computer Vision Basics

Lecture Outline

- Classes
- Cutting Edge Graphics (SIGGRAPH)
- Ray Tracing Optimization
- Computer Vision Basics



Overview
Overview Within Area
Single Topic
Deep Dive
Related/Helpful
CS 148
CS 44N
CS 248B
CS 248A
CS 348C
CS 348E ?
CS 348K
EE 267 (W)
CS 448Z
CS 123
CS 448I
ME 339
CS 448B
CS 231N
CS 205L

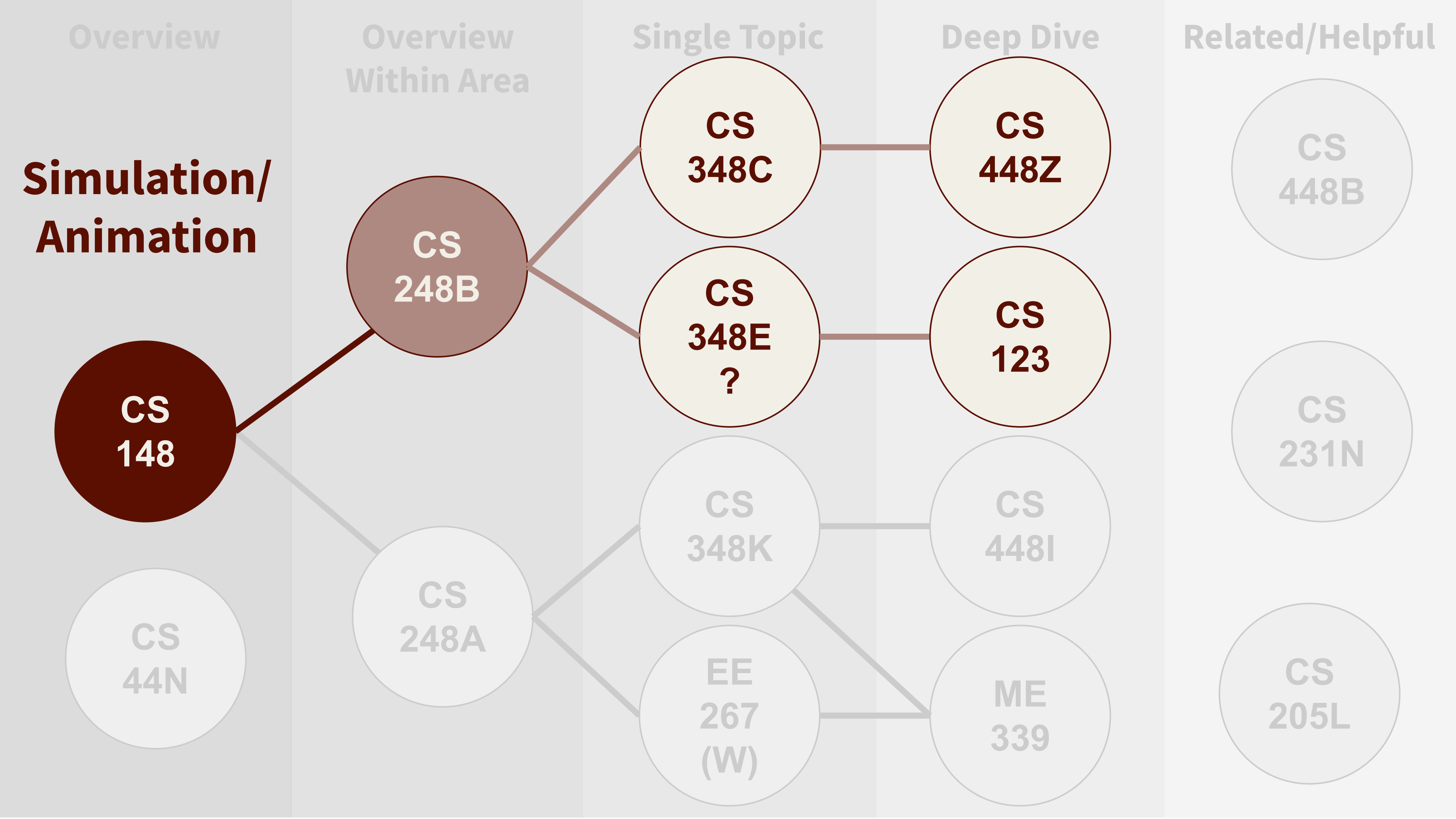

Overview
Overview Within Area
Single Topic
Deep Dive
Related/Helpful
CS 148
CS 44N
CS 248B
CS 248A
CS 348C
CS 348E ?
CS 348K
EE 267 (W)
CS 448Z
CS 123
CS 448I
ME 339
CS 448B
CS 231N
CS 205L

Fall Quarter

CS 44N: Great Ideas in Computer Graphics

A hands-on interactive and fun exploration of great ideas from computer graphics. Motivated by graphics concepts, mathematical foundations and computer algorithms, students will explore an eccentric selection of "great ideas" through short weekly programming projects. Project topics will be selected from a diverse array of computer graphics concepts and historical elements.

Prof. Doug James



Overview
Overview Within Area
Single Topic
Deep Dive
Related/Helpful
Simulation/ Animation
CS 148
CS 44N
CS 248B
CS 248A
CS 348C
CS 348E ?
CS 348K
EE 267 (W)
CS 448Z
CS 123
CS 448I
ME 339
CS 448B
CS 231N
CS 205L

Fall Quarter

CS 248B: Fundamentals of Computer Graphics: Animation and Simulation

- Learn more about simulation and motion
- Focus on sound and motion
- Rigid body and soft body interactions
- Particle systems
- IK bones (rigging)

Prof. Karen Liu

Prof. Doug James

Fall Quarter

CS 248B: Fundamentals of Computer Graphics: Animation and Simulation

- Learn more about simulation and motion
- Focus on sound and motion
- Rigid body and soft body interactions
- Particle systems
- IK bones (rigging)

Winter Quarter

CS 348C: Computer Graphics: Animation and Simulation

- Simulation and Animation, but focused on particle systems
- Taught in Houdini
- Particle systems, procedural geometry, dynamics, and sound
- Final project

Prof. Doug James

Winter Quarter

CS 348C: Computer Graphics: Animation and Simulation

- Simulation and Animation, but focused on particle systems
- Taught in Houdini
- Particle systems, procedural geometry, dynamics, and sound
- Final Project

Spring Quarter

Prof. Doug James

CS 448Z: Physically Based Animation and Sound

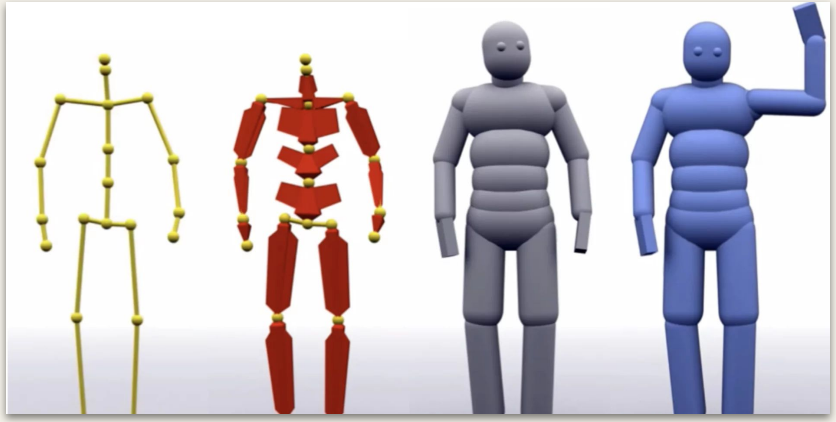
- Modeling sound waves and radiation
- Sound propagation and acoustic transfer
- Fracture
- Fire
- Cloth

Spring Quarter

CS 348E: Character Animation: Modeling, Simulation, and Control of Human Motion

- Simulation and Animation, but focused on the human body
- Forward and Inverse Kinematics
- Study of hands
- Using motion to predict future actions

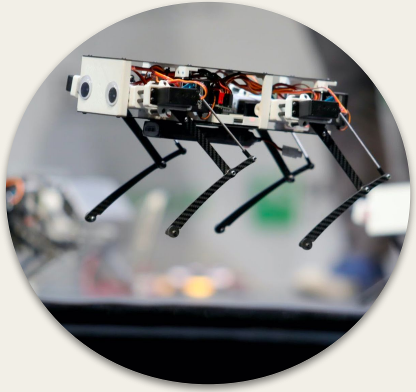
Prof. Karen Liu



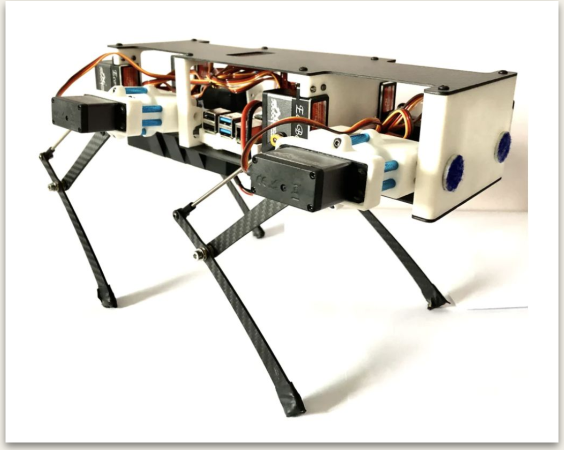
Fall and Spring Quarter

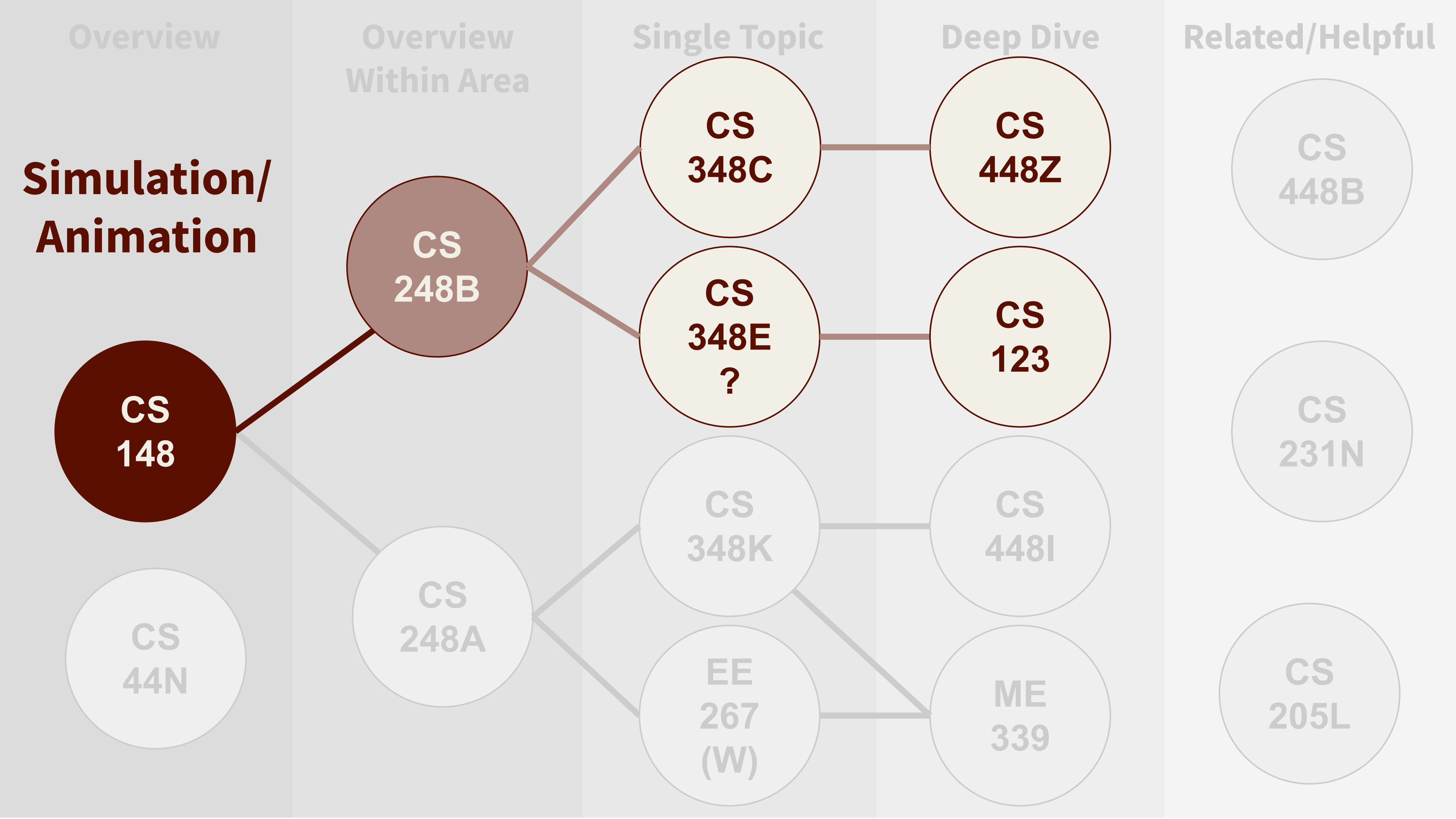
CS 123: A Hands-On Introduction to Building AI-Enabled Robots

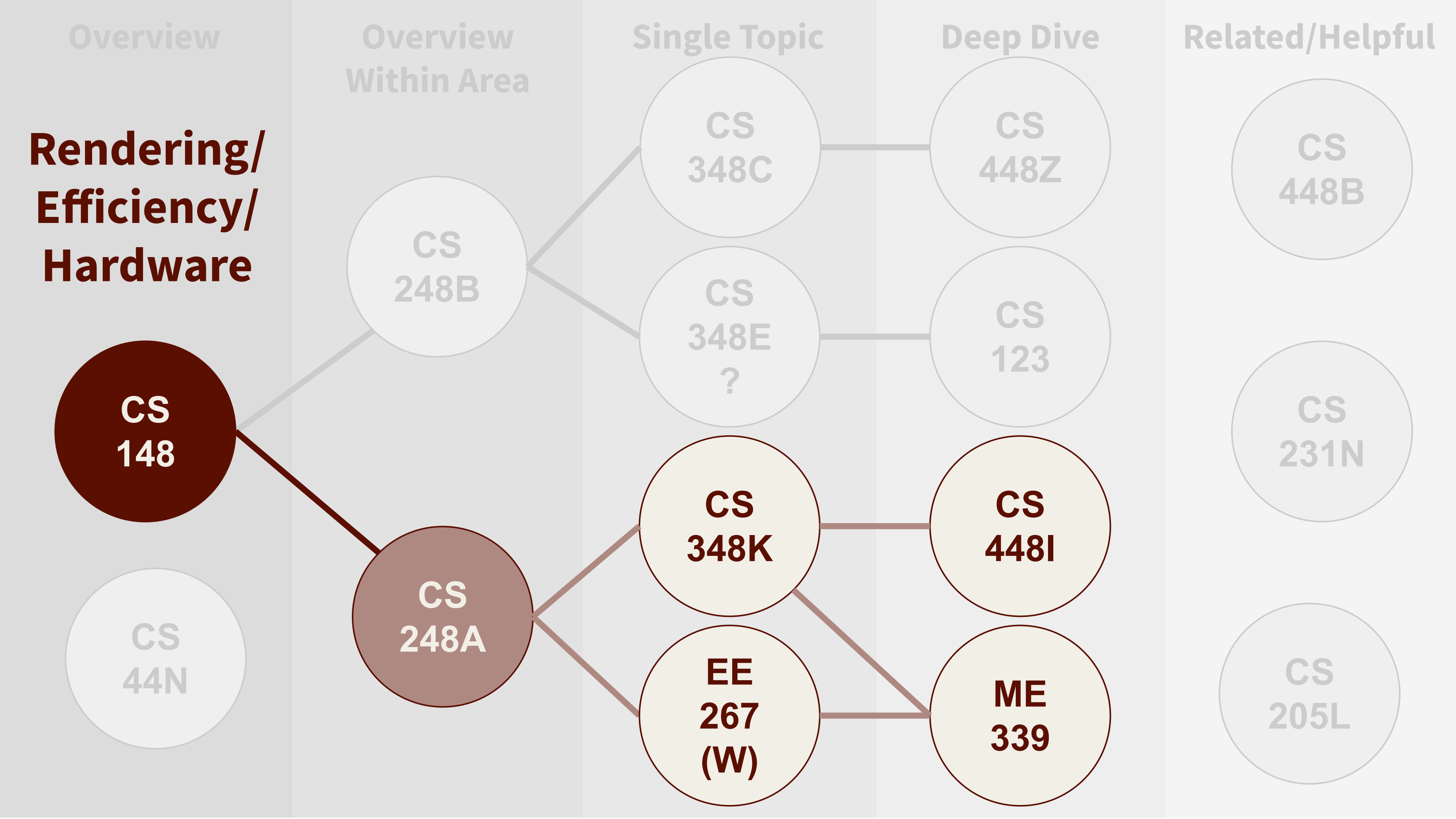
- Build your own quadruped robot
- Program the Robot
- Requires only basic linear algebra and python
- Only 20 slots



Prof. Karen Liu

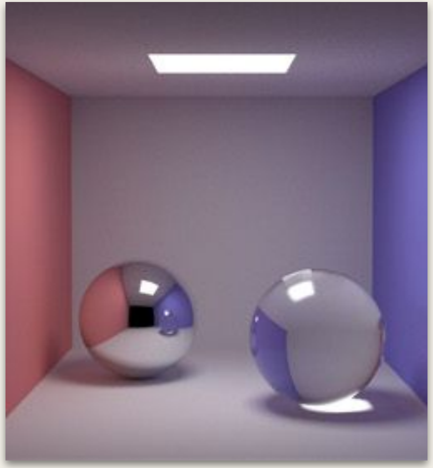



Overview
Overview Within Area
Single Topic
Deep Dive
Related/Helpful
Simulation/ Animation
CS 148
CS 44N
CS 248B
CS 248A
CS 348C
CS 348E ?
CS 348K
EE 267 (W)
CS 448Z
CS 123
CS 448I
ME 339
CS 448B
CS 231N
CS 205L

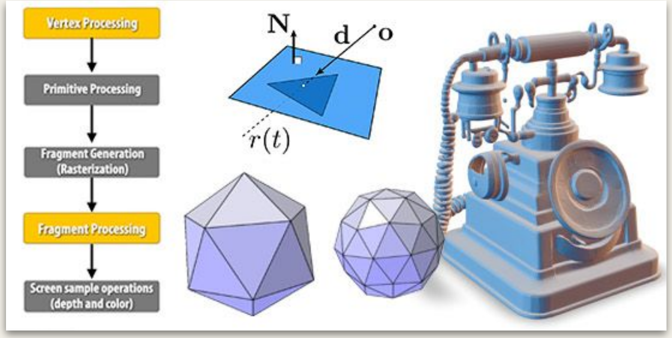

Overview
Overview Within Area
Single Topic
Deep Dive
Related/Helpful
Rendering/ Efficiency/ Hardware
CS 148
CS 44N
CS 248B
CS 248A
CS 348C
CS 348E ?
CS 348K
EE 267 (W)
CS 448Z
CS 123
CS 448I
ME 339
CS 448B
CS 231N
CS 205L

CS 248A: Computer Graphics: Rendering, Geometry, and Image Manipulation

- What is an image and how does it get shown on the screen?
- GPU pipeline
- Transformations
- A bit of interaction
- Final Project



Prof. Kayvon Fatahalian



CS 348K: Visual Computing Systems

- Digital camera processing
- Neural network hardware
- Video compression
- NERFS
- GPU accelerated ray tracing

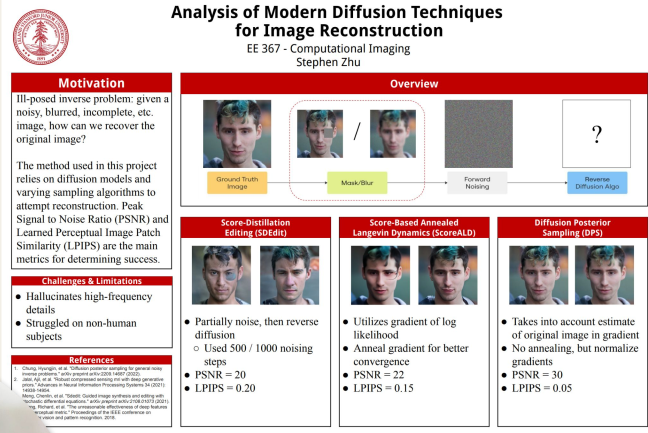
Prof. Kayvon Fatahalian

```
Func blur_3x3(Func input) {
  Func blur_x, blur_y;
  Var x, y, xi, yi;

  // The algorithm - no storage or order
  blur_x(x, y) = (input(x-1, y) + input(x
  blur_y(x, y) = (blur_x(x, y-1) + blur_x

  // The schedule - defines order, locali
  blur_y.tile(x, y, xi, yi, 256, 32)
        .vectorize(xi, 8).parallel(y);
  blur_x.compute_at(blur_y, x).vectorize(

  return blur_y;
}
```

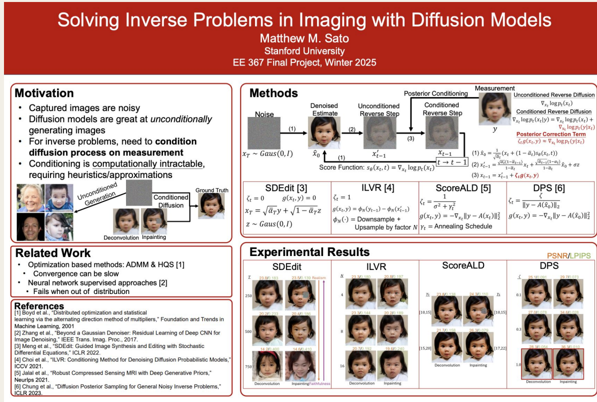
Spring Quarter

CS 448I: Computational Imaging

- Human Perception and Digital Cameras
- Using ML to solve inverse problems (denoising, etc.)
- Light field imaging
- Wave optics



Prof. Gordon Wetstein

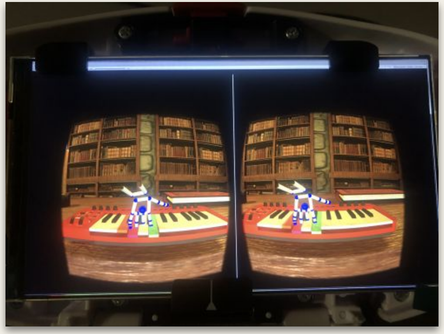


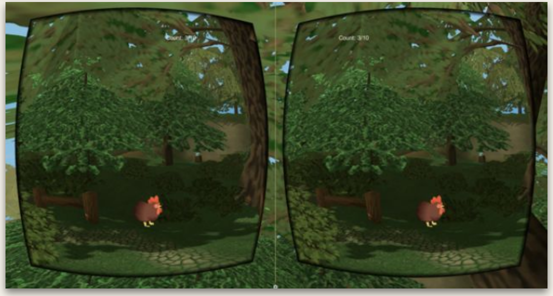
Spring Quarter

EE 267: Virtual Reality

- Hands on electrical engineering class
- Hardware systems used for virtual reality
- Write code and explore software for VR
- Project based class

Prof. Gordon Wetstein



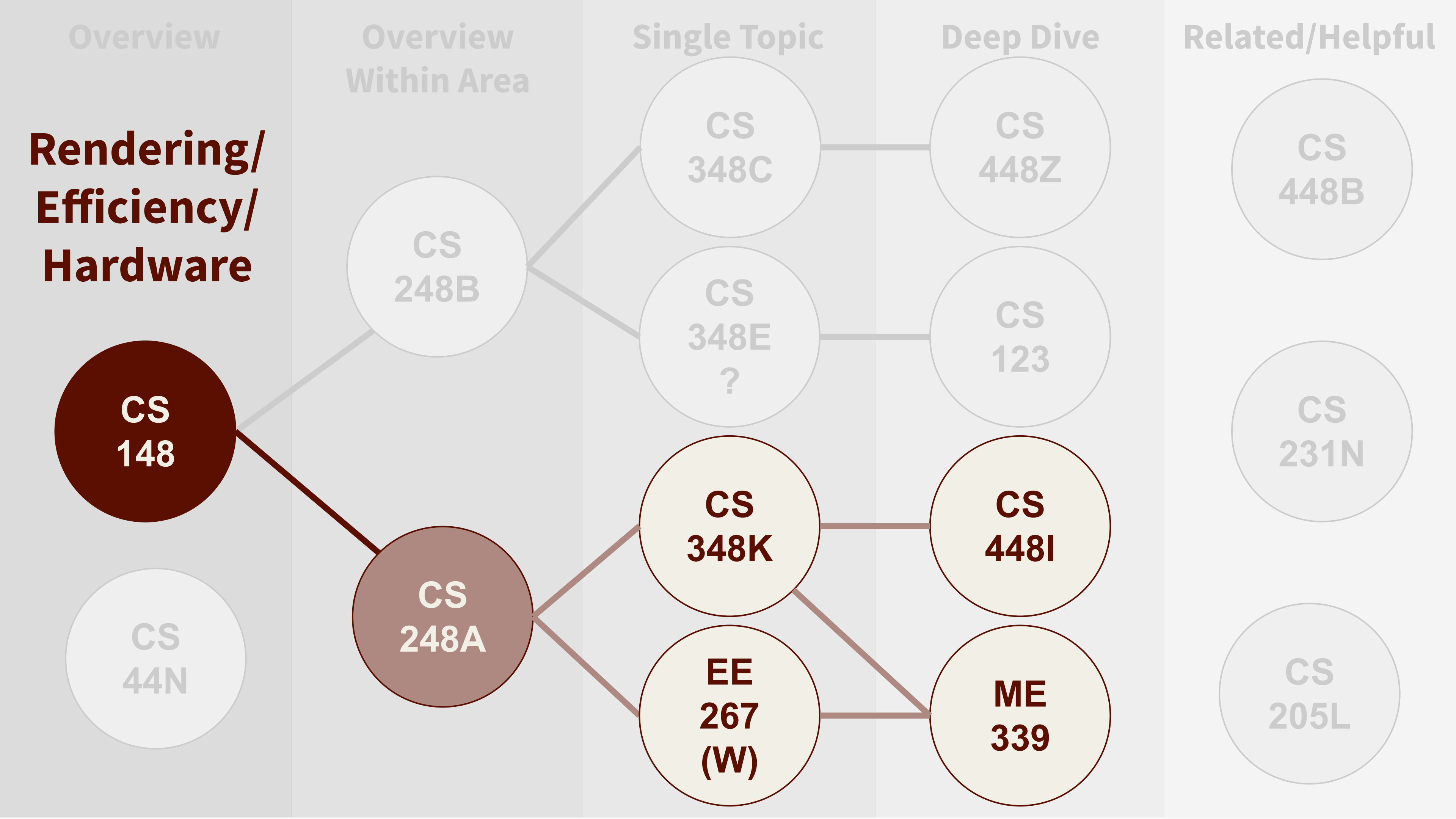


Spring Quarter

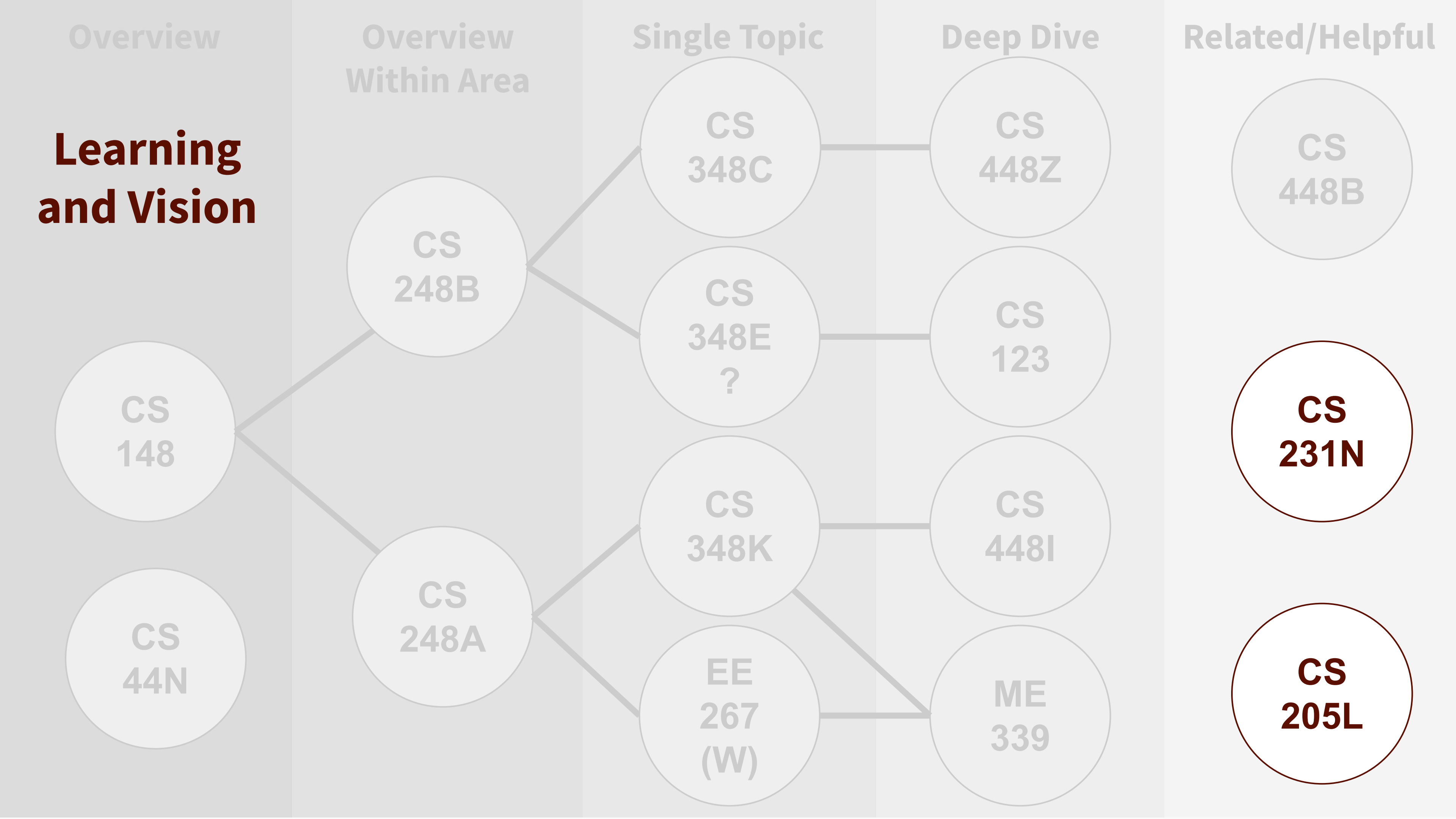
ME 339: Introduction to Parallel Computing Using MPI, OpenMP and CUDA

- Programming multicore processors, GPUs, and parallel computers
- Multi-threaded programs
- Machine learning

Prof. Eric Darve



Overview
Overview Within Area
Single Topic
Deep Dive
Related/Helpful
Rendering/ Efficiency/ Hardware
CS 148
CS 44N
CS 248B
CS 248A
CS 348C
CS 348E ?
CS 348K
EE 267 (W)
CS 448Z
CS 123
CS 448I
ME 339
CS 448B
CS 231N
CS 205L



Overview
Overview Within Area
Single Topic
Deep Dive
Related/Helpful
Learning and Vision
CS 148
CS 44N
CS 248B
CS 248A
CS 348C
CS 348E ?
CS 348K
EE 267 (W)
CS 448Z
CS 123
CS 448I
ME 339
CS 448B
CS 231N
CS 205L

Winter Quarter

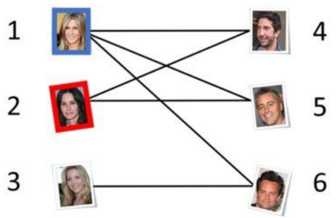
CS 205L: Continuous Mathematical Methods with an Emphasis on Machine Learning

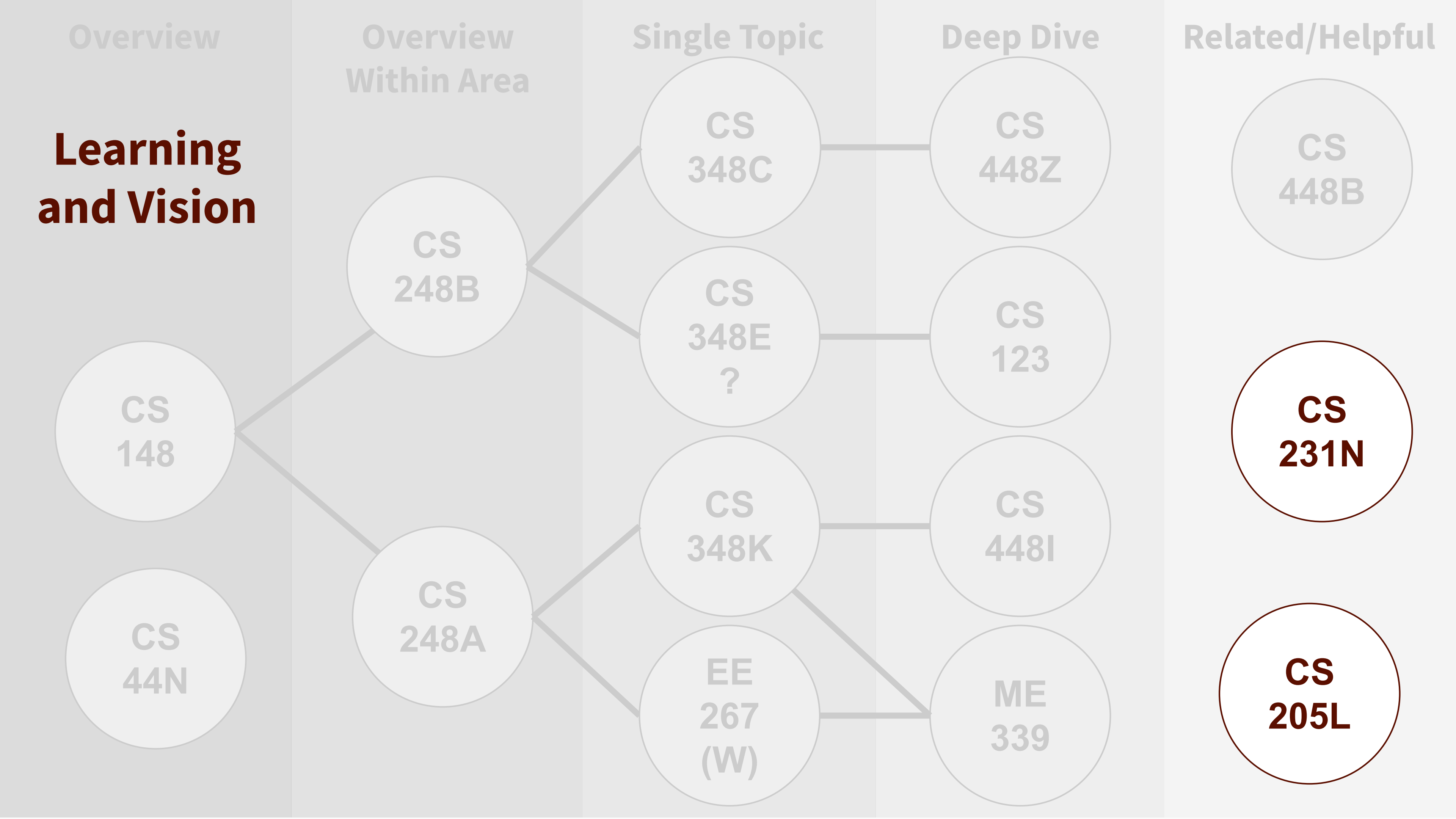
- Learn about the mathematical foundation of Machine Learning
- Basis in Linear Algebra
- Programming and paper assignments

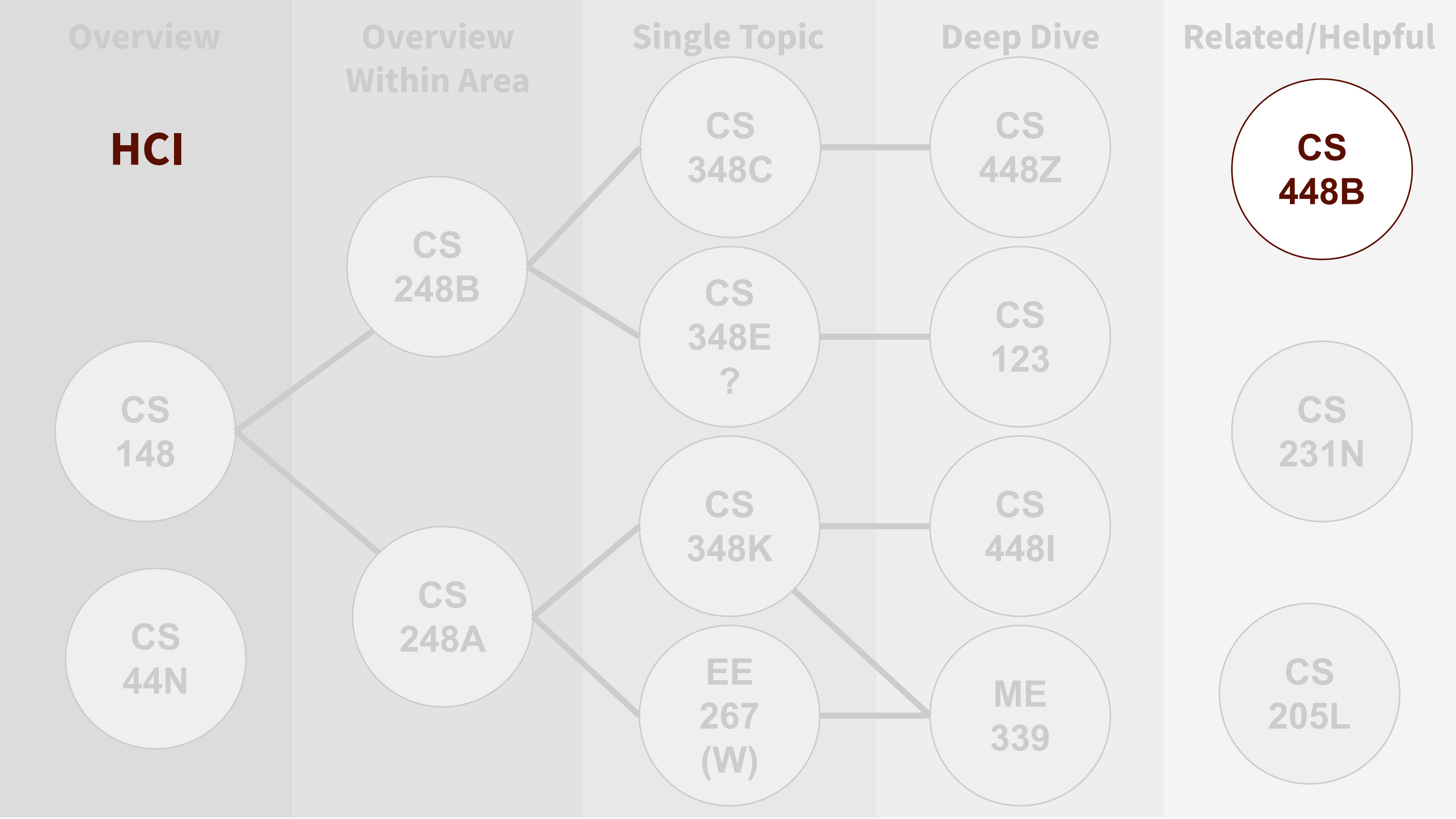
Prof. Ron Fedkiw

Adjacency Matrix

$$\begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{array}
\begin{array}{c} \begin{array}{cccccc} 1 & 2 & 3 & 4 & 5 & 6 \end{array} \\ \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \end{array}$$



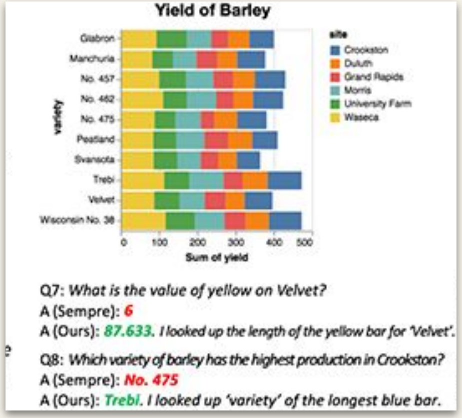

Overview
Overview Within Area
Single Topic
Deep Dive
Related/Helpful
Learning and Vision
CS 148
CS 44N
CS 248B
CS 248A
CS 348C
CS 348E ?
CS 348K
EE 267 (W)
CS 448Z
CS 123
CS 448I
ME 339
CS 448B
CS 231N
CS 205L

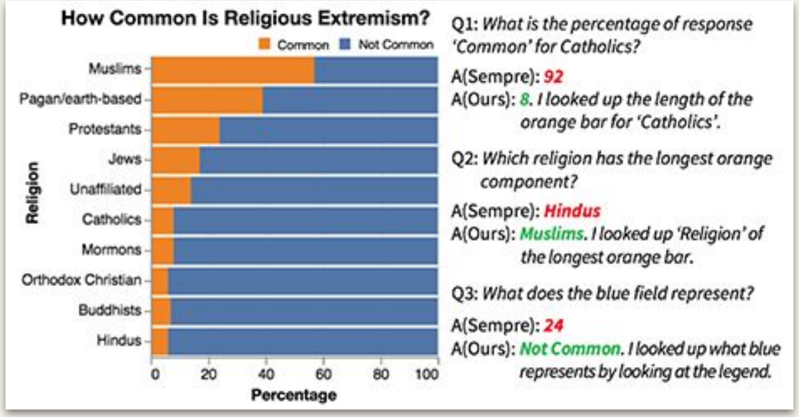

Overview
Overview Within Area
Single Topic
Deep Dive
Related/Helpful
HCI
CS 448B
CS 248B
CS 348C
CS 448Z
CS 348E ?
CS 123
CS 148
CS 231N
CS 348K
CS 448I
CS 44N
CS 248A
EE 267 (W)
ME 339
CS 205L

CS 448B: Data Visualization

- Think about Computer Graphics from a comprehension perspective
- Learn how to use and optimize visualization tools
- Discuss data visualization theory

Prof. Maneesh Agrawala





MI 260: Creative Visualization Studio

BIOE 281: Biomechanics of Movement (ME 281)

ARTSTUDI 167M: Animated By Origins: Africa and The Americas (AFRICAAM 167)

BIOE 485: Modeling and Simulation of Human Movement (ME 485)

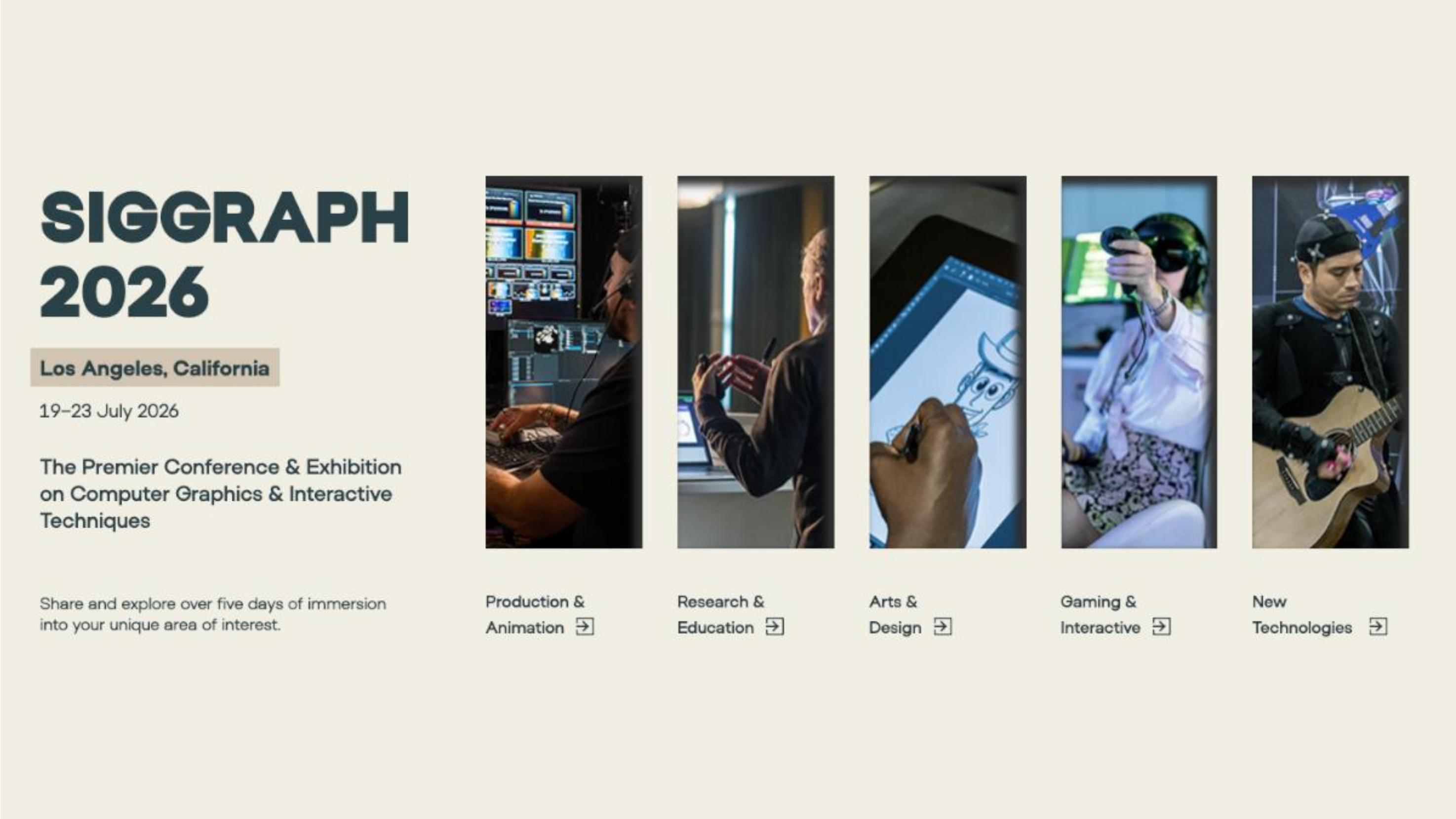
AMSTUD 129: Animation and the Animated Film (FILMEDIA 129, 329, 429)

PSYCH 230: Computational Models of Human Creative Expression

Lecture Outline

- Classes
- Cutting Edge Graphics (SIGGRAPH)
- Ray Tracing Optimization
- Computer Vision Basics

SIGGRAPH


SIGGRAPH 2026
Los Angeles, California
19–23 July 2026
The Premier Conference & Exhibition on Computer Graphics & Interactive Techniques
Share and explore over five days of immersion into your unique area of interest.
Production & Animation
Research & Education
Arts & Design
Gaming & Interactive
New Technologies

SIGGRAPH: Papers

https://gfx.cs.princeton.edu/pubs/Barnes_2009_PAR/index.php

PatchMatch: A Randomized Correspondence Algorithm for Structural Image Editing (2009)

- Nearest-neighbor searches for variety of high level digital image editing tools

(a) original (b) hole+constraints (c) hole filled (d) constraints (e) constrained retarget (f) reshuffle

Figure 1: *Structural image editing. Left to right: (a) the original image; (b) a hole is marked (magenta) and we use line constraints (red/green/blue) to improve the continuity of the roofline; (c) the hole is filled in; (d) user-supplied line constraints for retargeting; (e) retargeting using constraints eliminates two columns automatically; and (f) user translates the roof upward using reshuffling.*

SIGGRAPH: Papers

https://ttwong12.github.io/papers/gestalt/gestalt.html

Conjoining Gestalt Rules for Abstraction of Architectural Drawings (2011)

- Leveraging psychology in graphics
- Gestalt = tendency of forming shapes/pictures with limited geometric information

Figure 1: *Simplification of a complex cityscape line-drawing obtained using our Gestalt-based abstraction.*

SIGGRAPH: Papers

https://www.dgp.toronto.edu/projects/cubic-stylization/

Cubic Stylization (2019)

- Purposely modeling more cartoon-y without making things too block-y
- Can tune the block-y parameter

Fig. 9. One can control the CUBENESS by changing the λ parameter in Eq. 1.

SIGGRAPH: Papers

https://repo-sam.inria.fr/fungraph/3d-gaussian-splatting/

3D Gaussian Splatting for Real-Time Radiance Field Rendering (2023)

- Useful for sparse points in camera calibration
- SOTA rendering and competitive training times

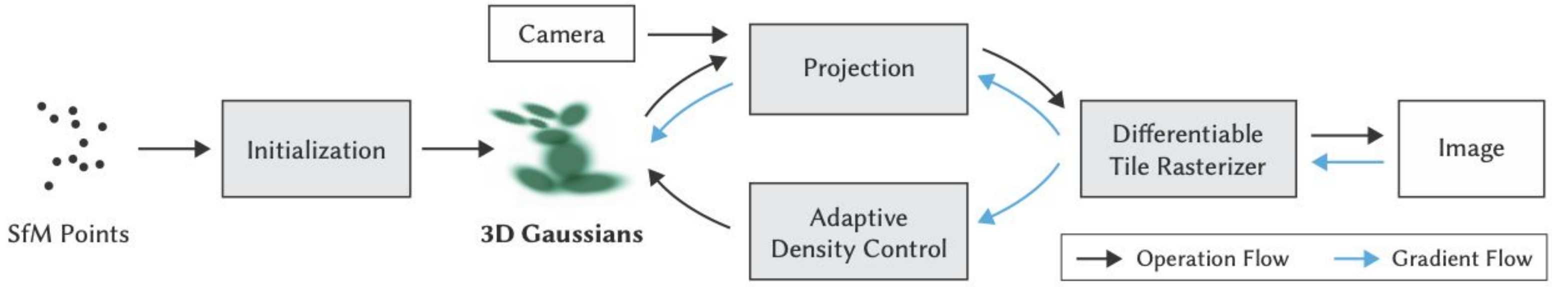


Fig. 2. Optimization starts with the sparse SfM point cloud and creates a set of 3D Gaussians. We then optimize and adaptively control the density of this set of Gaussians. During optimization we use our fast tile-based renderer, allowing competitive training times compared to SOTA fast radiance field methods. Once trained, our renderer allows real-time navigation for a wide variety of scenes.

SIGGRAPH: Papers

CLAY: A Controllable Large-scale Generative Model for Creating High-quality 3D Assets (2024)

- A 3D geometry and material generator designed to take human input and transform it into intricate 3D digital structures

https://dl.acm.org/doi/10.1145/3658146

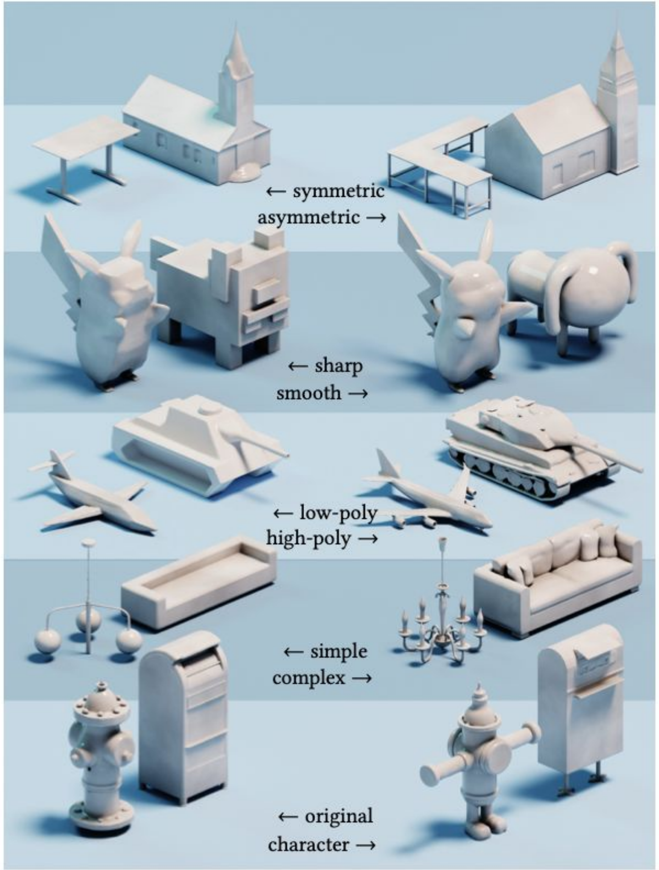


Fig. 11. Evaluation of the CLAY's ability to alter generated content by incorporating different geometric feature tags in the prompt. We showcase precise controls over the geometry style, in the extreme case transforming a fireplug into a T-pose character.

SIGGRAPH: Papers

DressCode: Autoregressively Sewing and Generating Garments From Text Guidance (2024)

- A text-driven 3D garment generation framework for fashion design, virtual try-on, and digital human creation

https://ihe-kaii.github.io/DressCode/

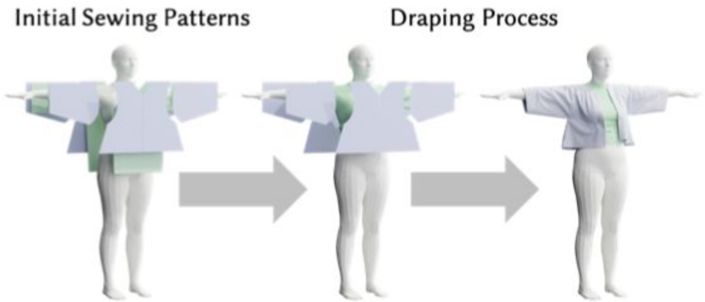


Fig. 6. **Examples of our multiple garments draping process.** Starting with initial sewing patterns, we first drape the inside T-shirt, followed by draping the outside jacket onto the model's body.

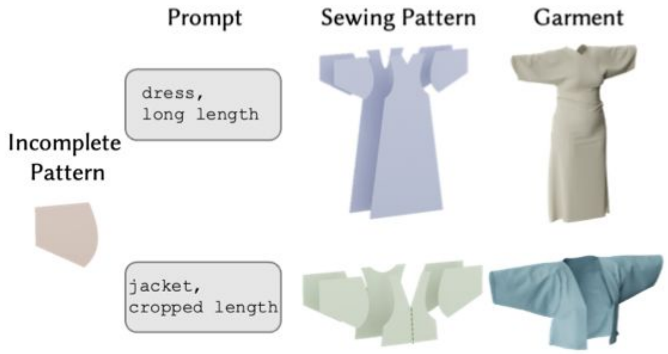


Fig. 7. **Examples of pattern completion.** Given an incomplete pattern, our method infers reasonable sewing pattern completions with various text prompts.

SIGGRAPH: Papers

https://dl.acm.org/doi/10.1145/3721238.3730724

Painting with 3D Gaussian Splatting Brushes (2025)

- Allows artists to directly paint 3D scenes using Gaussian splat brushes
- Create photorealistic effects

Fig. 1. Our method explores the creative affordances of painting with real-world texture-geometry content, represented as 3D Gaussian Splats. Our novel technique enables 3D artists to source both their brushes and their canvases from in-the-wild captures, and to swiftly remix highly realistic content through direct and controllable painting for fast 3D prototyping and ideation. Here we use a set of Gaussian splat brushes to embellish this small patch of a local park.

SIGGRAPH: Papers (2026)

https://arxiv.org/abs/2601.18585

GimmBO: Interactive Generative Image Model Merging via Bayesian Optimization

- Fine tuning customization of diffuse models used to create styles/subjects underrepresented in original training data
- Image generation with preferential bayesian optimization (human-in-the-loop)

SIGGRAPH: Papers (2026)

https://dl.acm.org/doi/pdf/10.1145/3811312

Mixwell: Sharp 2D Fluid Brushes for Progressive Physics-Based Mixing

- GPU accelerated methods for physics based mixing

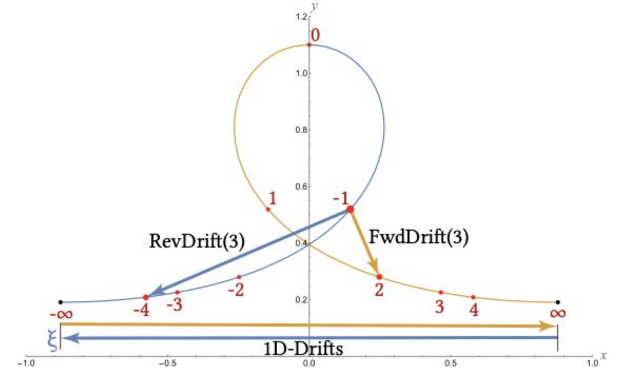


Fig. 6. **2D Drift Illustration:** The world-frame particle drift from Figure 2 annotated with (a) forward- and reverse-drift displacements of the point $\boldsymbol{p}(-1)$ (at time $t=-1$) for $\Delta t = \pm 3$; and (b) horizontal 1D drift displacements shown at the bottom for the reverse-drift $\xi(\eta_\infty) < 0$ in blue, and the opposite 1D forward drift $(-\xi)$ in orange.

Fig. 7. **Mixwell Brush Deformation:** A bird-like figure marbled on a modest HD-resolution raster using a real-time Shadertoy implementation. The active stroke accumulates its deformation map separately to avoid blurring the total deformation map. These maps are stored at screen resolution and interpolated at lookup, so quality is below our fully supersampled results.

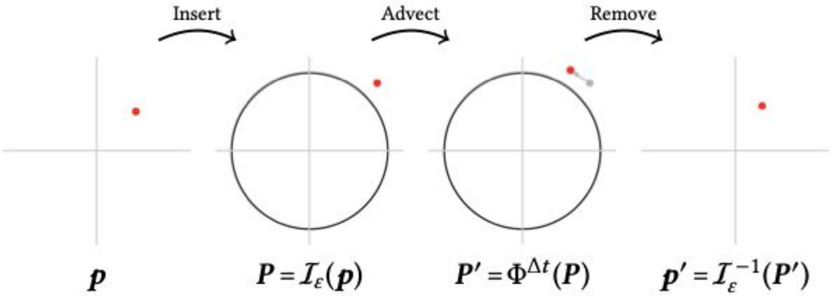


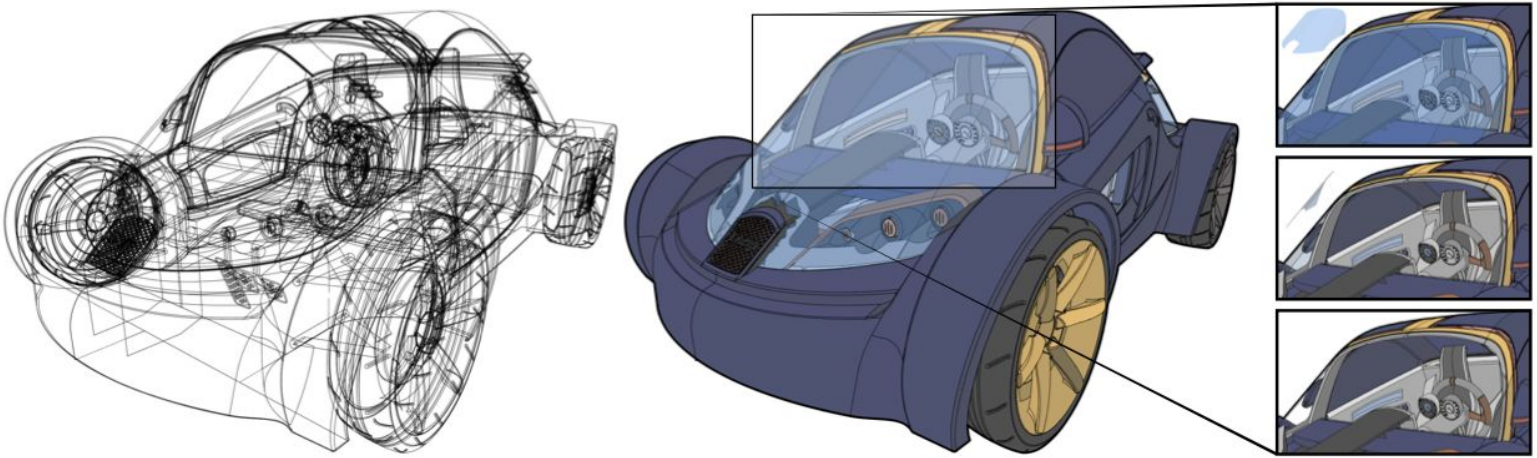
Fig. 8. **Mixwell Brush Construction**

SIGGRAPH: Papers (2026)

https://www.cs.cmu.edu/~kmcrane/Projects/RobustPlanarMaps/RobustPlanarMaps.pdf

Robust Planar Maps for 3D Vectorization

- Converting 3D scenes into clean 2D vector images
- Vectorizing 3D scenes with proper occlusion!

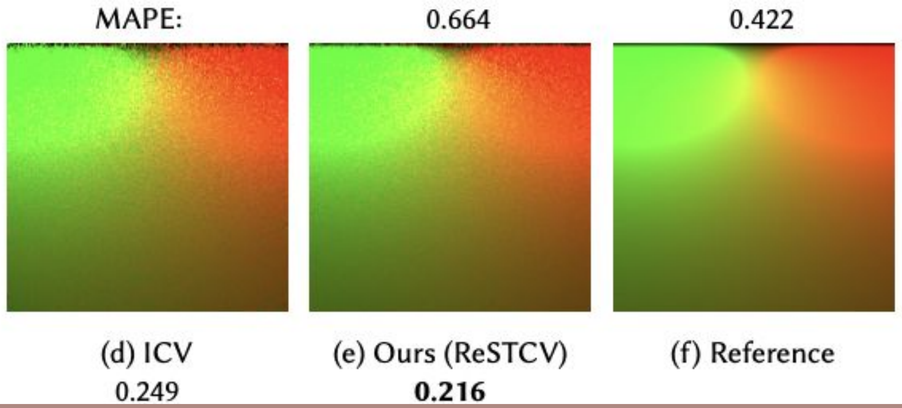


SIGGRAPH: Papers (2026)

https://dl.acm.org/doi/10.1145/3799902.3811113

Spatio-Temporal Control Variates with ReSTIR for Real-Time Rendering

- Path tracing, soft shadows, and global illumination at real time rates



SIGGRAPH: Papers (2026)

https://dl.acm.org/doi/10.1145/3799902.3811066

EchoAvatar: Real-time Generative Avatar Animation from Audio Streams

- Real time avatar animation driven by audio
- Jointly trained on music-dance and speech-gesture datasets

Fig. 1. Given streaming audio input, our method generates avatar animation in a streaming manner. The four poses shown above are sampled from a continuous motion sequence driven by the audio stream.

SIGGRAPH: Papers (2026)

https://dl.acm.org/doi/10.1145/3799902.3811042

gCDT: A Highly Parallel GPU Algorithm for Large-Scale Constrained Delaunay Triangulation

- Useful for CAD and scientific visualization
- Flip propagation to let a GPU thread perform short process and gets down number of global synchronization rounds

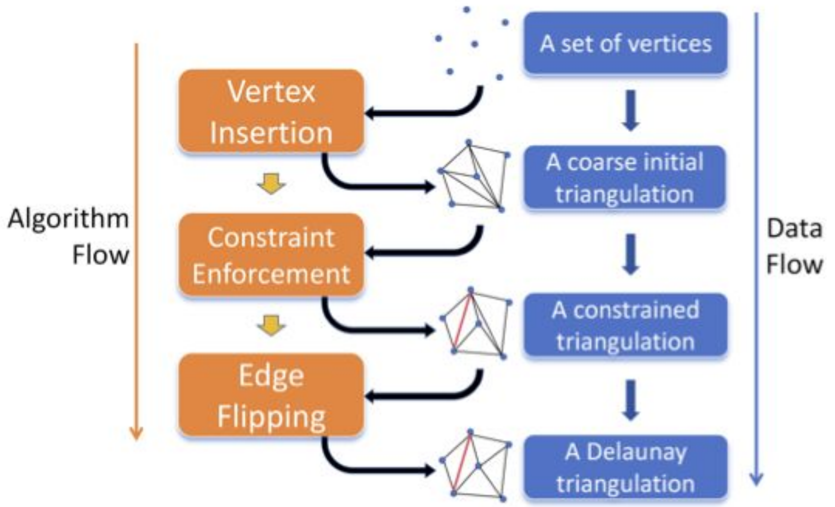


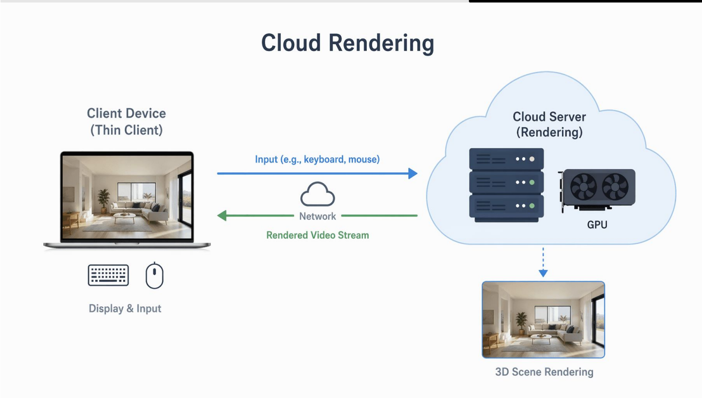
Fig. 2. **Algorithm workflow.** Overview of the workflow, showing each stage and the data structures it updates.

Lecture Outline

- Classes
- Cutting Edge Graphics (SIGGRAPH)
- Ray Tracing Optimization
- Computer Vision Basics

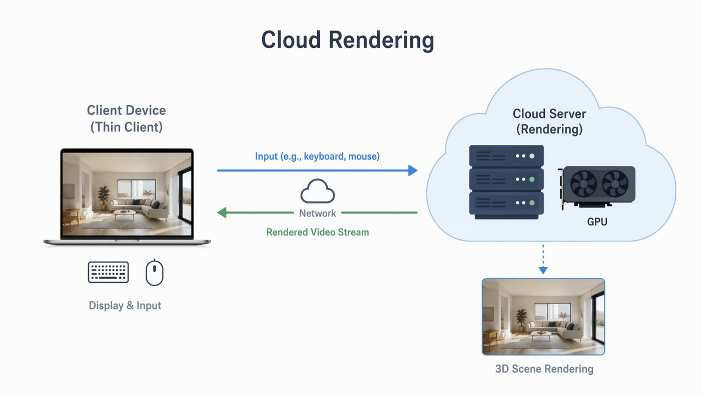
The Future of Ray Tracing

- Real time rendering accessibility (for **VR and AR**)



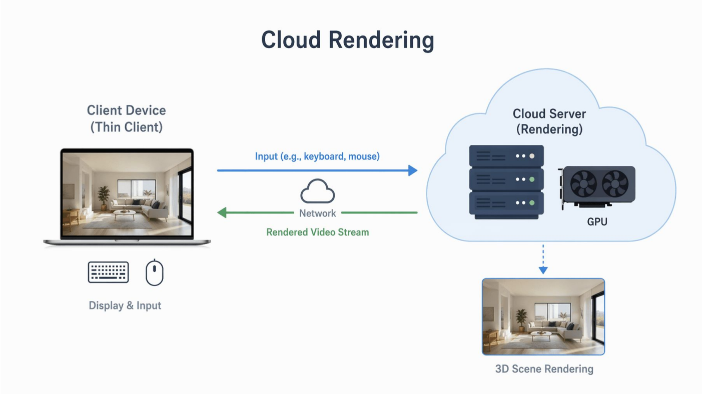
The Future of Ray Tracing

- Real time rendering accessibility (for **VR and AR**)
- Wider adoption by **game engines**
  - Unreal and Unity are integrating this tech natively



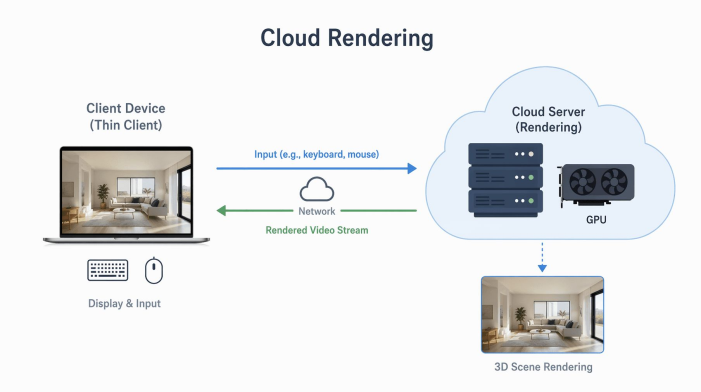
The Future of Ray Tracing

- Real time rendering accessibility (for **VR and AR**)
- Wider adoption by **game engines**
  - Unreal and Unity are integrating this tech natively
- **Hybrid rendering**
  - Combines rasterization and ray tracing
  - E.g. Eevee Next!



The Future of Ray Tracing

- Real time rendering accessibility (for **VR and AR**)
- Wider adoption by **game engines**
  - Unreal and Unity are integrating this tech natively
- **Hybrid rendering**
  - Combines rasterization and ray tracing
  - E.g. Eevee Next!
- **Cloud rendering**
  - Expanding access to ray tracing to allow users to experience its benefits



Ray Tracing Optimization: Progress

Ray Tracing Optimization: Progress

NVIDIA RTX 20 Series

- NVIDIA first generation of ray tracing GPUs (2018)
- RT cores designed specifically to accelerate ray tracing calculations
- Tensor cores for denoising and smart upscaling
- Introduced **DLSS** (deep learning super sampling)

Ray Tracing Optimization: Progress

NVIDIA RTX 30 Series

- Tensor cores for ray tracing
- Delivered more realistic lighting, shadows, and reflections

NVIDIA RTX 40 Series

- Ray tracing overdrive for improving accuracy
- Used for Cyberpunk and Fortnite for realism
- Power efficiency and thermal performance as a priority

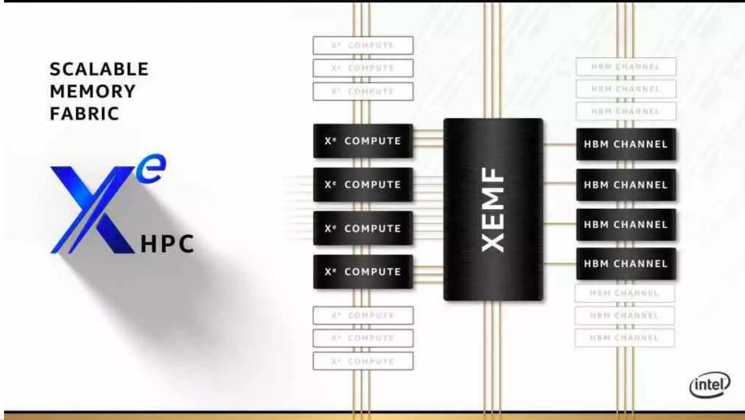
Ray Tracing Optimization: Progress

AMD Radeon RX 6000 Series

- Real time ray tracing
- Emphasis on power efficiency for gaming and professional workloads
- AMD FidelityFX Super Resolution
  - Open source upscaling tech to boost performance

Intel Xe Architecture

- Great for modern gaming



Ray Tracing Optimization: AI-Powered Games

Ray Tracing Optimization: AI-Powered Games

TensorRT for RTX Plugin

- Runtime for Unreal Engine's NNE (neural network engine)
- AI deployment in real time applications (1.5x performance to direct ML)
- Path traced hair in Unreal Engine 5.7!

Ray Tracing Optimization: AI-Powered Games

TensorRT for RTX Plugin

- Runtime for Unreal Engine’s NNE (neural network engine)
- AI deployment in real time applications (1.5x performance to direct ML)
- Path traced hair in Unreal Engine 5.7!

NVIDIA Kimodo

- Generates high quality human motion to fill gaps in animation
- Trained on huge dataset of 3D motion capture data

Comfy UI

- Image generation, 3D object generation, etc.

Lecture Outline

- Classes
- Cutting Edge Graphics (SIGGRAPH)
- Ray Tracing Optimization
- Computer Vision Basics

Computer Vision: Basics

- **Computer Graphics**
  - Typically 3D → 2D
  - Generate images inspired by the real world
- **Computer Vision**
  - Typically 2D → 3D understanding
  - Understand the real world through images

Computer Vision: Basics

- **Computer Graphics**
  - Typically 3D → 2D
  - Generate images inspired by the real world
- **Computer Vision**
  - Typically 2D → 3D understanding
  - Understand the real world through images
- Huge part of our brains are dedicated to visual processing – difficult to replicate in a computer
- Huge research area
- Synthetic data generation

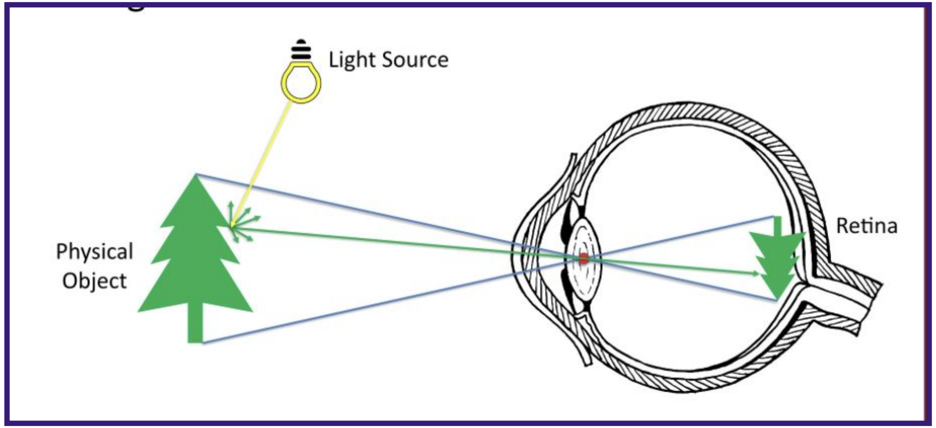
Computer Vision: Basics

Low-level Vision	Geometric Vision	Visual Recognition	Advanced Topics
Filters	Image formation	Machine Learning	Video Understanding
Edges	Perspective	Segmentation	3D Deep Learning
Features	Camera models	Clustering	
Matching	Multiview geometry		

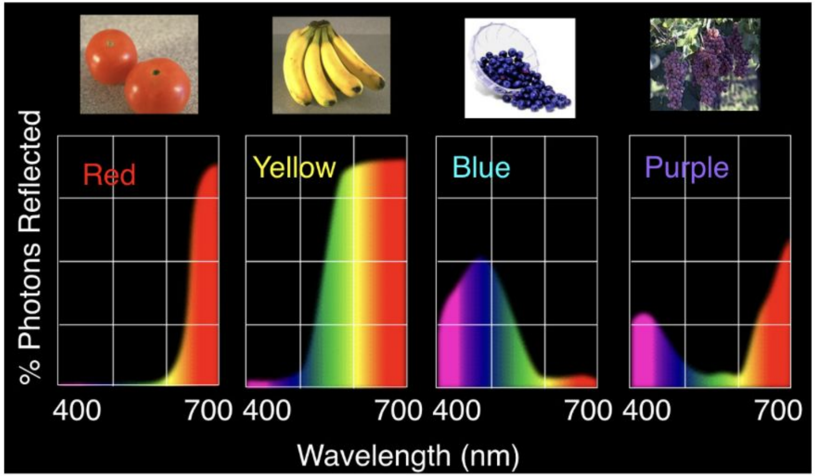
https://stanford-cs131.github.io/spring2026/

Computer Vision: Basics

Understanding images



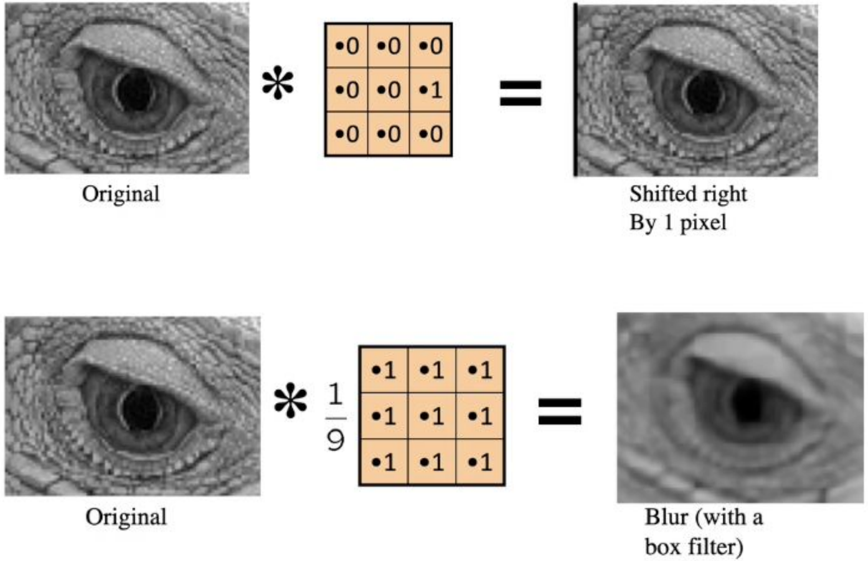
Some examples of the reflectance spectra of surfaces

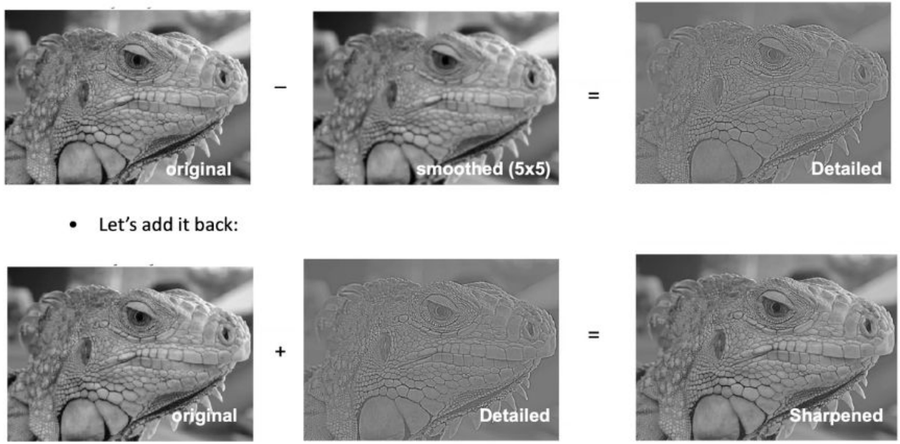


https://stanford-cs131.github.io/spring2026/

Computer Vision: Convolutions

- Using small matrices (**kernels**) to modify images





https://stanford-cs131.github.io/spring2026/

Computer Vision: Filters

- Using small matrices (**kernels**) to modify images

2D moving average over a 3 × 3 window of neighborhood

$$g[n,m] = \frac{1}{9} \underbrace{\sum_{k=n-1}^{n+1}}_{\text{Sum over rows}} \underbrace{\sum_{l=m-1}^{m+1}}_{\text{Sum over columns}} f[k,l]$$

$$= \frac{1}{9} \sum_{k=-1}^{1} \sum_{l=-1}^{1} f[n-k, m-l]$$

2D moving average over a 3 × 3 neighborhood window

Original image

Smoothed image

$f[n,m]$

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	0	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

$g[n,m]$

	0	10	20						

https://stanford-cs131.github.io/spring2026/

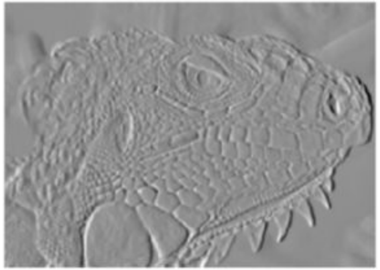
Computer Vision: Edges

- Using small matrices (**kernels**) to modify images

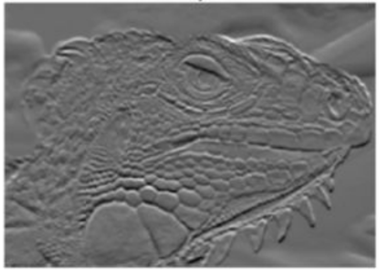
3x3 image gradient filters

$$\frac{1}{3}\begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix} \qquad \frac{1}{3}\begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}$$

Derivative in x direction



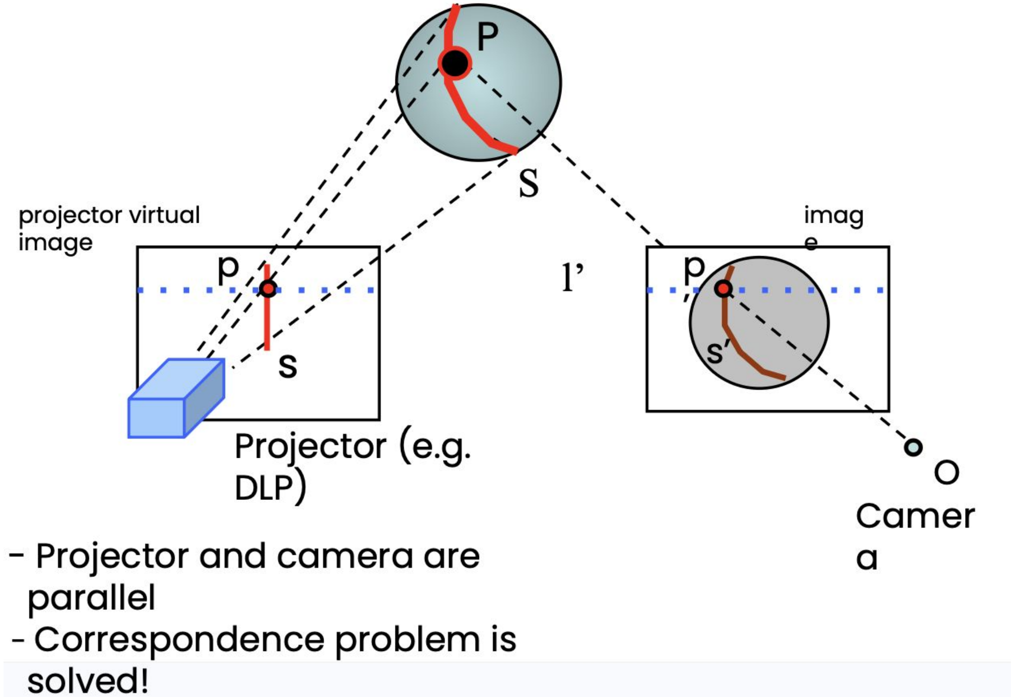
Derivative in y direction



https://stanford-cs131.github.io/spring2026/

Computer Vision: Calibration

- System calibration required to understand 3D world

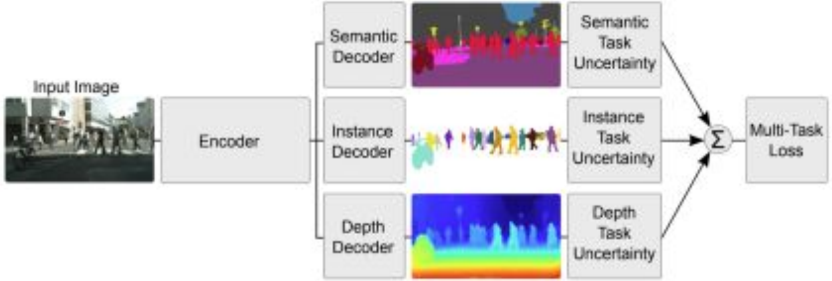


https://stanford-cs131.github.io/spring2026/

Computer Vision: AI / ML

- Learn from (many) fully labeled examples $(\boldsymbol{x}_i, \boldsymbol{y}_i)$
- Dominant paradigm in most applications
- Examples: classification (discrete y), regression (continuous y)
- Multi-task Learning: learn multiple (related) tasks jointly $(\boldsymbol{x}_i, \{\boldsymbol{y}_i^{1:K}\})$
- Few-shots Learning: learning from small labeled datasets (e.g., one-shot)

Classification: cat or dog?
Regression: what weight?



Multi-Task Learning Using Uncertainty to Weigh Losses for Scene Geometry and Semantics, Kendall et al, CVPR'18

https://stanford-cs131.github.io/spring2026/

Computer Vision: CS131 Final Project

SurveilMate: Tracking Chores between Roommates

Kate Baker
Stanford University

Yasmine Alonso
Stanford University

Abstract

Dishes are a household task that often gets left to the last moment, even as organized roommates and close friends. To address this issue, we developed a tool that uses facial and object recognition to track which roommate adds a dirty dish to the sink and who subsequently cleans it. We leveraged an MIT licensed facial recognition library that uses facial detection, alignment, and identification. We used `YOLOv8s` *for object recognition and openCV for region of interest capture and video processing. The resulting tool was performant in simple tests (e.g. one dish added at a time) but struggled when adding stacked dishes or many small dishes at a time. Improvements include having a robust dataset with photos of our (stacked) dishes and transforming our tool into a real-time dish tracker.*

Computer Vision: CS131 Final Project

SurveilMate: Tracking Chores between Roommates

Kate Baker
Stanford University

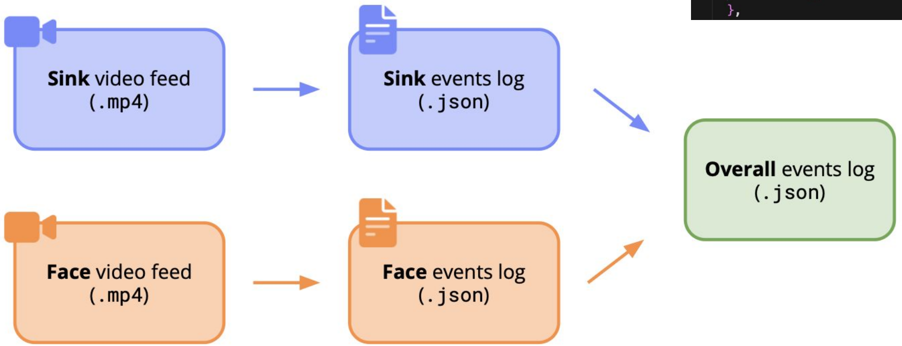
Yasmine Alonso
Stanford University

Abstract

Dishes are a household task that often gets left to the last moment, even as organized roommates and close friends. To address this issue, we developed a tool that uses facial and object recognition to track which roommate adds a dirty dish to the sink and who subsequently cleans it. We leveraged an MIT licensed facial recognition library that uses facial detection, alignment, and identification. We used `YOLOv8s` *for object recognition and openCV for region of interest capture and video processing. The resulting tool was performant in simple tests (e.g. one dish added at a time) but struggled when adding stacked dishes or many small dishes at a time. Improvements include having a robust dataset with photos of our (stacked) dishes and transforming our tool into a real-time dish tracker.*

- ageitgey's model built using dlib's SOTA face recognition deep learning algorithm
- Applies Histograms of Oriented Gradients (HOG) for human detection
  - HOG leverages CS131 topics!
  - Convolution, kernels, edge information

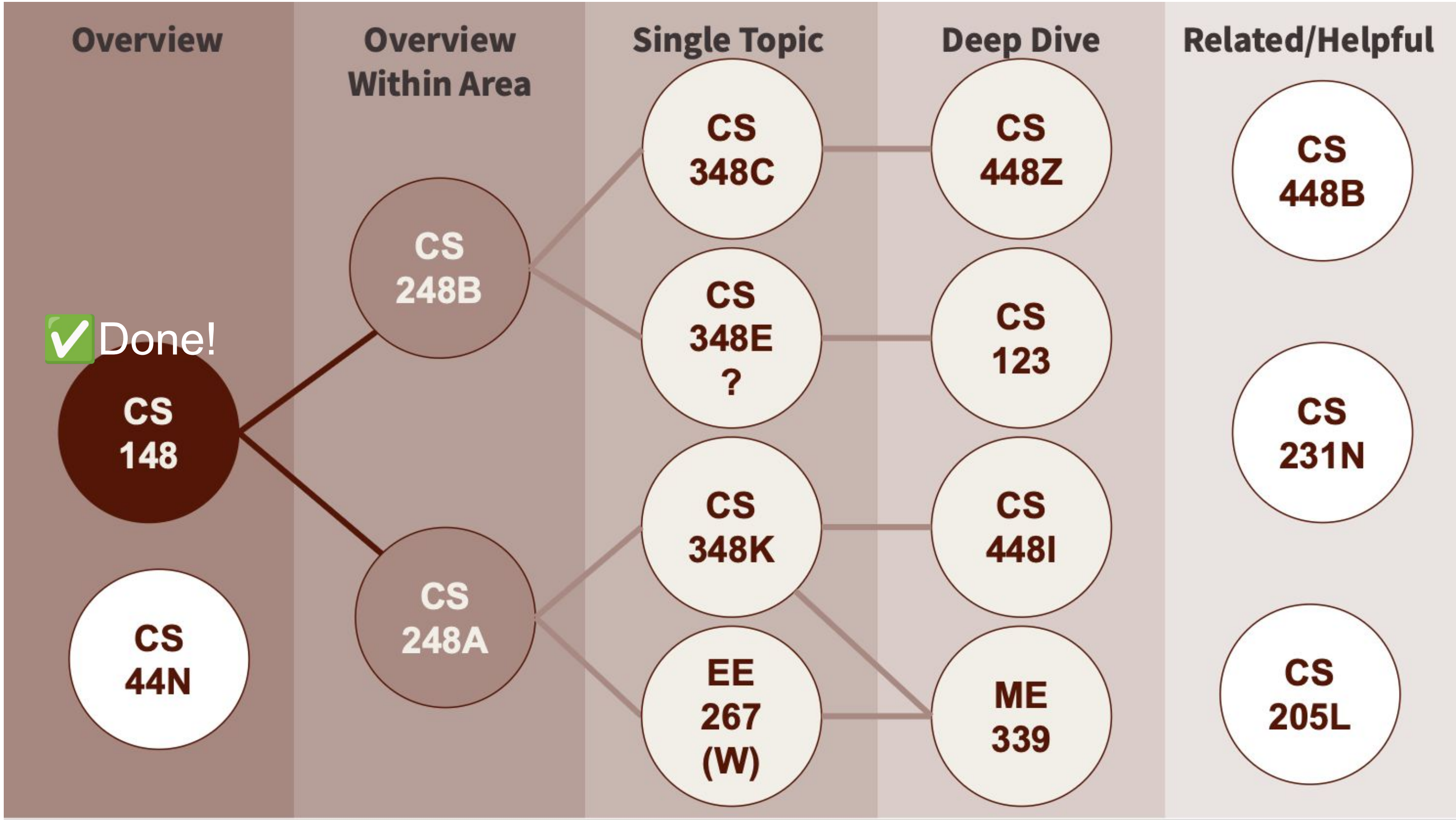
```
[
  {
    "name": "Yasmine Alonso",
    "entered": "0:0",
    "left": "0:00:01.33",
    "entered_seconds": 0.0,
    "left_seconds": 1.33,
    "duration_seconds": 1.33
  },
  {
    "name": "Yasmine Alonso",
    "entered": "0:00:03.67",
    "left": "0:00:04.83",
    "entered_seconds": 3.67,
    "left_seconds": 4.83,
    "duration_seconds": 1.17
  },
```



If you’re interested in industry…

- Make a demo reel
- Meet people in the field
    - Volunteer for SIGGRAPH!
    - Look for and go to events
- Take project based classes
- Save your work!

Yay!!


Overview
Overview Within Area
Single Topic
Deep Dive
Related/Helpful
✅Done!
CS 148
CS 44N
CS 248B
CS 248A
CS 348C
CS 348E ?
CS 348K
EE 267 (W)
CS 448Z
CS 123
CS 448I
ME 339
CS 448B
CS 231N
CS 205L

Attendance… for the last time :(

https://tinyurl.com/272ru3m2