Computer Vision

Guide to AI and Data Courses · Part III: Signals, Images and Audio

Meaning from images | Features | Recognition and detection | Cameras and 3D | Video and motion | Related courses

Sercan KÜLCÜ · Giresun University · Computer Engineering

© Sercan Külcü. All rights reserved. · Interactive HTML version of these slides: https://sercankulcu.github.io/files/ai_courses/slides/Chapter_18_Computer_Vision.html

INTRODUCTION

What Question Does It Ask?

> **How does a computer understand what is in an image or a video?**

Computer vision studies how to **extract meaning** from an image or video: which objects are there, where they are, what 3D structure the scene has. The input is an image but the output is not; it is a label, a box, a mask or depth. Transforming the image is the job of image processing.

At a glance

APPROACH extracting meaning from images	**INPUT** image or video	**OUTPUT** label, box, depth, 3D structure
SUCCESS MEASURE accuracy, mAP, IoU	**MATH LOAD** high: geometry, linear algebra	**DATA NEEDS** many labeled images

A Short History

1963
Larry Roberts's PhD thesis recovers 3D blocks from line drawings: the field's first work.

1966
MIT's "Summer Vision Project" expects to solve vision in one summer; it does not.

1982
Marr's book Vision appears, describing vision as a layered computational process.

1999
Lowe introduces SIFT features: local point matching invariant to scale and rotation.

2001
Viola and Jones publish a real-time face detector; it ships in digital cameras.

2012
AlexNet wins ImageNet by a wide margin; the deep learning era in vision begins.

THE COURSE

Core Concepts

Local features
Corners and keypoints: Harris, SIFT, ORB.

Classification
One label for the whole image: "cat".

Object detection
A box and label per object: YOLO, Faster R-CNN.

Semantic segmentation
A class for every pixel: road, pedestrian, car.

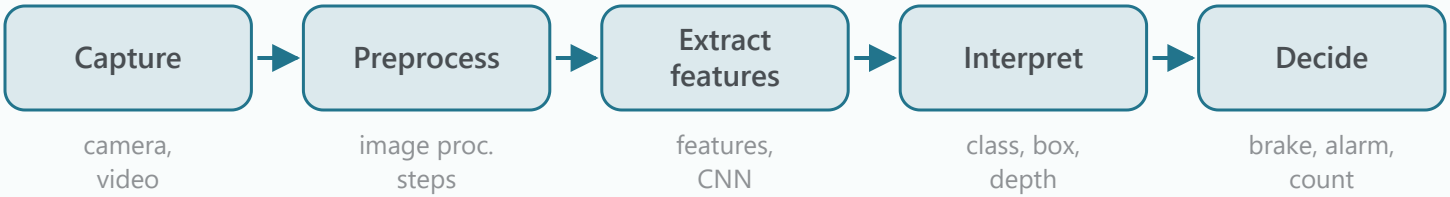
Camera model
Pinhole model, calibration, intrinsics and extrinsics.

Multi-view geometry
Stereo, epipolar geometry, structure from motion.

Motion and tracking
Optical flow; following an object frame to frame.

Convolutional networks
CNNs and vision transformers (ViT).

A Typical Workflow



Example: A driver assistance system grabs a frame from the front camera, balances brightness, extracts features with a convolutional network and marks pedestrians with boxes; it also estimates their distance from a stereo camera. The output is not an image but the fact "pedestrian at 8 meters"; the system issues a brake warning accordingly.

THE COURSE

What Does the Course Cover?

1. Image formation and the camera model
2. Filters and edge detection (image processing review)
3. Local features: corners, SIFT, ORB
4. Feature matching and RANSAC
5. Homographies and panorama stitching
6. Camera calibration and stereo vision
7. Structure from motion and multi-view geometry
8. Optical flow and object tracking
9. Classification with convolutional networks
10. Object detection: the R-CNN family, YOLO
11. Semantic and instance segmentation
12. Vision-language models and current trends

Skills you will gain

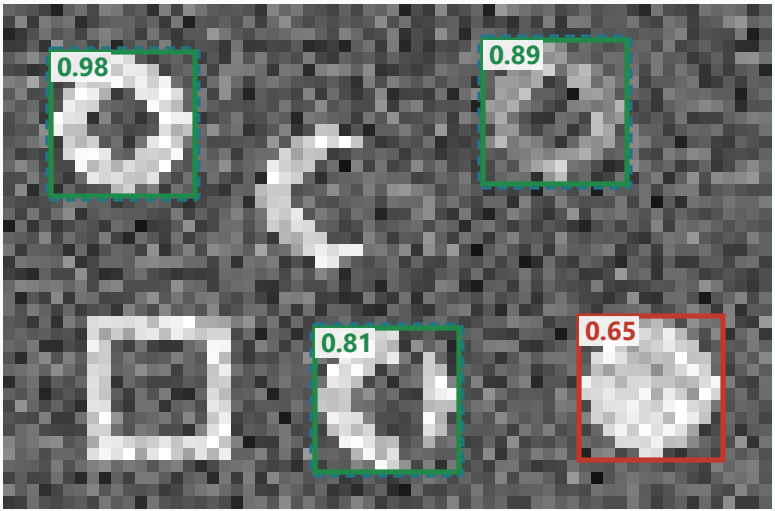
OpenCV | using and fine-tuning pretrained models in PyTorch | camera calibration | preparing labeled data | evaluating with IoU and mAP

A typical semester; the order and coverage vary between universities.

TRY IT

Find the Rings in the Image

On an assembly line a camera must count the ring-shaped gaskets on the belt. A template slides over every position and similarity is measured. Change the confidence threshold and suppression, and check the boxes.





12×12 template

Confidence threshold = **0.62**

Non-maximum suppression | Score map

1643 windows scanned, above threshold **15**
After suppression: 4 boxes
Correct: **3** False: **1** Missed: **0**
Mean IoU: 1.00

Dashed blue: ground truth. Green: correct detection, red: false.

Image processing transforms the image; computer vision draws a **decision** from it: box, label, confidence. The course teaches how to measure detection with IoU, true/false detections and threshold choice.

CONTEXT

Prerequisites

Courses worth knowing before this one, and why they are needed.

Course	Why it is needed
Image Processing	edges, filters, segmentation
Linear algebra	matrices, eigenvalues and projection: the language of camera geometry
Programming	array operations and using libraries in Python
Algorithms	search, matching and complexity: feature matching and tracking

CONTEXT

Are You Ready for This Course?

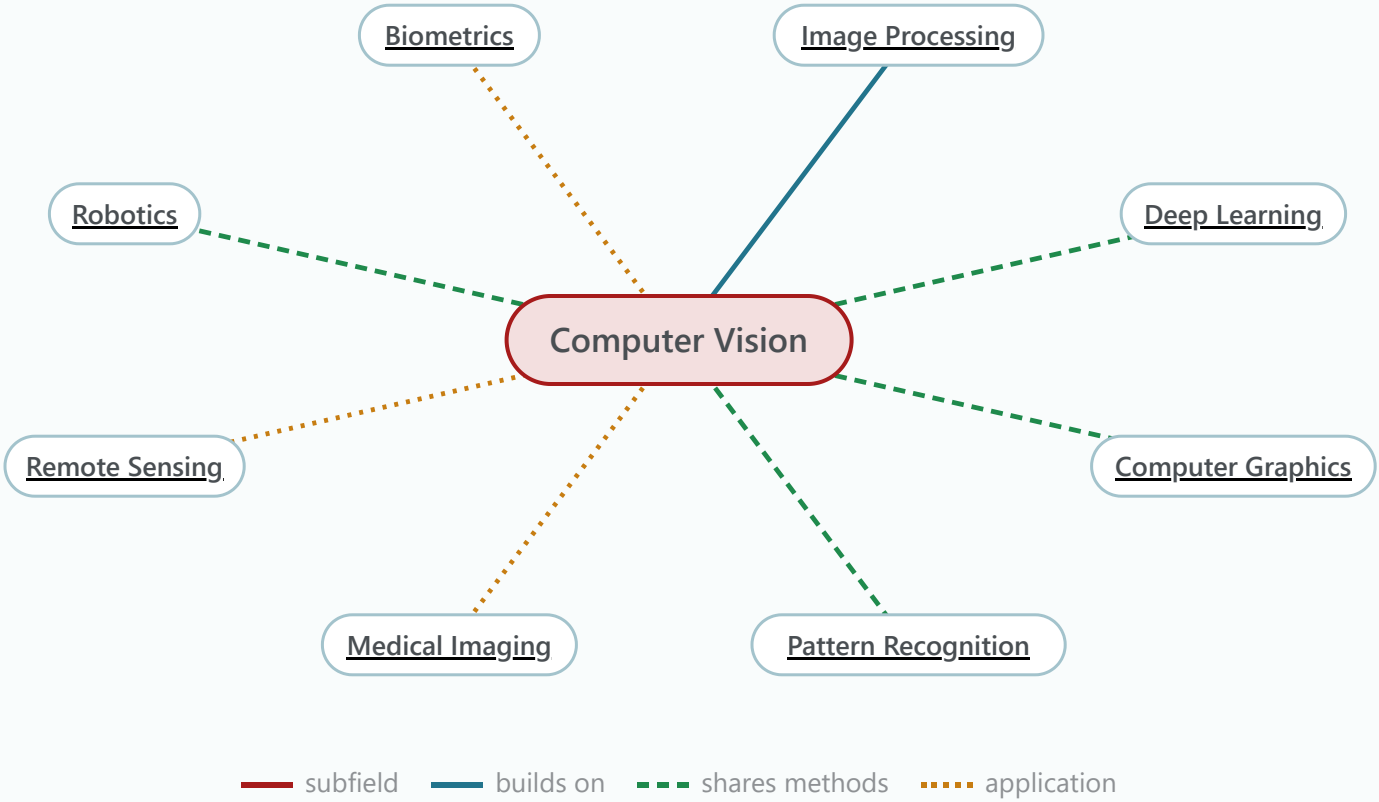
Ask yourself; for every "no", the right-hand column shows where to start.

Ask yourself	If not, start with
Can you explain matrix multiplication, inverses and eigenvalues?	Linear Algebra
Do you know that an image is a matrix of pixels, and what convolution does?	Image Processing
Can you explain how a neural network is trained with a loss and gradient descent?	Artificial Neural Networks
Can you work with NumPy arrays in Python?	An intro programming course, then a short NumPy tutorial

If you can say "yes" to most of these, you are ready; one or two gaps can be closed during the semester.

CONTEXT

Neighbouring Courses



Overlapping Courses

Relation	Course	Shared ground
builds on	This course builds on Image Processing	edges, filters, segmentation
application	Medical Image Processing applies the methods of this course	lesion detection, classification
application	Remote Sensing applies the methods of this course	land cover classification, change detection
application	Biometric Systems applies the methods of this course	face recognition
shares methods	Shares methods with Deep Learning	convolutional neural networks, vision transformers
shares methods	Shares methods with Pattern Recognition	object classification
shares methods	Shares methods with Computer Graphics	camera model; graphics renders scenes, vision recovers them
shares methods	Shares methods with Robotics and Robot Vision	robot vision, SLAM

DIFFERENCES

Where It Differs (1/2)

Criterion	Computer Vision	Image Processing	Deep Learning
Core question	How does a computer understand what is in an image or a video?	How is an image enhanced, transformed and divided into regions?	How do many-layered networks learn useful representations from raw data?
Approach	extracting meaning from images	pixel-level transformations	multilayer representation learning
Input	image or video	image	raw data: images, text, audio
Output	label, box, depth, 3D structure	processed image or regions	learned representation and prediction
Success measure	accuracy, mAP, IoU	visual quality, PSNR	accuracy, loss, scaling
Math load	high: geometry, linear algebra	medium: convolution, Fourier	high: derivatives, matrices, probability
Data needs	many labeled images	little; learning not required	large; needs GPUs

- **Image Processing:** In image processing both input and output are images; in computer vision the input is an image and the output is meaning: a label, a box, depth.
- **Deep Learning:** Deep learning is a general tool; vision uses it together with camera geometry, multi-view and motion knowledge.

DIFFERENCES

Where It Differs (2/2)

Criterion	Computer Vision	Computer Graphics	Pattern Recognition
Core question	How does a computer understand what is in an image or a video?	How is a realistic image produced from the description of a scene?	How do we decide from measurements which class an object belongs to?
Approach	extracting meaning from images	geometry and light simulation	statistical decision theory
Input	image or video	scene: models, lights, camera	measurement or feature vector
Output	label, box, depth, 3D structure	image or animation	class label
Success measure	accuracy, mAP, IoU	realism, frame rate	classification error
Math load	high: geometry, linear algebra	high: linear algebra, geometry	medium-high: probability, linear algebra
Data needs	many labeled images	none	labeled examples

- **Computer Graphics:** Graphics produces an image from a scene description; vision does the reverse, rebuilding the scene from the image.
- **Pattern Recognition:** Pattern recognition classifies any kind of measurement; vision treats classification together with spatial structure, position and 3D.

PRACTICE

Applications

Autonomous driving
Detecting lanes, pedestrians and traffic signs.

Face recognition
Phone unlock, access control.

Retail
Checkout-free stores, shelf monitoring.

Augmented reality
Locating the camera and adding objects to the scene.

Industry
Spotting defective products and removing them from the line.

Sports analytics
Tracking players and the ball, statistics.

PRACTICE

Risks and Ethics

Face recognition and surveillance

Identifying people in a crowd can turn into tracking without consent; many countries restrict its use in public spaces.

Unequal performance

Error rates can be markedly higher for skin tones and age groups that are under-represented in the training data.

Adversarial examples

A small sticker the human eye ignores can make a model read a stop sign as a different sign.

Safety-critical failures

In self-driving, a single missed pedestrian is a cost that average accuracy does not show.

In depth: Explainable AI and AI Ethics.

PRACTICE

Tools and Datasets

Tools and libraries

OpenCV | PyTorch / torchvision | Ultralytics YOLO | Detectron2 | COLMAP | Open3D

Well-known datasets

ImageNet | COCO | PASCAL VOC | KITTI | Cityscapes | ADE20K

Unlike image processing, vision needs **many labeled images**; in COCO the box and mask of every object were drawn by hand.

PRACTICE

Common Misconceptions

Common belief	In fact
Computer vision is another name for image processing.	Image processing transforms the image; vision extracts meaning from it. Vision uses image processing as a first step.
Vision is a solved problem.	Networks still fail noticeably under lighting, weather or cameras that differ from the training data.
Vision is just deep learning.	Calibration, stereo, SLAM and 3D reconstruction rest largely on geometry.
A model "sees" an object the way a human does.	Networks can over-rely on texture and context; a small pixel change can flip a decision.

NEXT STEPS

After This Course

Courses that build on or extend this one

Medical Image Processing | Remote Sensing | Biometric Systems

Related careers

Computer vision engineer | Autonomous systems engineer | AR/VR developer | Robotics perception engineer | Researcher

Computer vision is the center of the third part: robotics, biometrics, medical image processing and remote sensing apply its methods to their own domains. To go deeper, continue with deep learning and generative models; on the geometry side, with computer graphics and robotics.

NEXT STEPS

Further Reading

- Szeliski, *Computer Vision: Algorithms and Applications*, 2nd ed., Springer, 2022.
- Hartley and Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed., Cambridge University Press, 2004.
- Prince, *Computer Vision: Models, Learning, and Inference*, Cambridge University Press, 2012.
- Marr, *Vision*, W. H. Freeman, 1982.

The End

Chapter 18: Computer Vision

← Previous chapter · Map of the field · Next chapter →