What is Computer Vision?

• *Vision* is about discovering from images what is present in the scene and where it is.

• In *Computer Vision* a camera (or several cameras) is linked to a computer. The computer interprets images of a real scene to obtain information useful for tasks such as navigation, manipulation and recognition.
The goal of computer vision

What we see

What a computer sees

Source: S. Narasimhan
What is Computer Vision NOT?

- **Image processing**: image enhancement, image restoration, image compression. Take an image and process it to produce a new image which is, in some way, more desirable.

- **Computational Photography**: extending the capabilities of digital cameras through the use of computation to enable the capture of enhanced or entirely novel images of the world. (See my other course)
Why study it?

• Replicate human vision to allow a machine to see:
  – Central to that problem of Artificial Intelligence
  – Many industrial applications

• Gain insight into how we see
  – Vision is explored extensively by neuroscientists to gain an understanding of how the brain operates (e.g. the Center for Neural Science at NYU)
Applications

- Intelligent machines (AI)
- Industrial inspection
e.g. light bulbs, electronic circuits
- Automotive
- Security
e.g. facial recognition in airports
- Image/video retrieval
- Digital cameras are everywhere now….

Huge number of startups & companies, e.g.
240 @ CVPR2017 conference:

3D Reconstruction

Pollefeys et al.

Goesele et al.
Handwritten Digit Recognition

Digit recognition, AT&T labs
Prof. Yann LeCun (NYU)

License plate readers
http://en.wikipedia.org/wiki/Automatic_number_plate_recognition

1/3 of all checks written in US use this system

Source: S. Seitz
Convolutional Neural Networks

• LeCun et al. 1989
• Neural network with specialized connectivity structure
• Huge revival in 2012
Goal

- Image Recognition
  - Pixels $\rightarrow$ Class Label

[Krizhevsky et al. NIPS 2012]
ImageNet Classification (2010 – 2015)
Object Detection Progress

![Graph showing the progress of object detection over years with markers for 'Before deep convnets' and 'Using deep convnets'.]
Mask R-CNN [He et al. 2017]
Mask R-CNN [He et al. 2017]
Face Detection

- Real-time face detection on most phones/cameras now
- Use to set exposure
- Also input for face recognition system
Face Recognition

- Used by Facebook, Google etc.
- Tag people’s faces in photos
- Need to distinguish a person’s face from many others

[Taigman et al. DeepFace: Closing the Gap to Human-Level Performance in Face Verification, CVPR’14]
Advanced Photo Search

• Text-based image search
  – (that actually looks at image)

Self-Driving Cars

- **Mobileye**: Vision systems in high-end BMW, GM, Volvo models
- Plans for self-driving by 2020

Source: A. Shashua
Self-Driving Cars

• Many other companies:
  – Uber
  – Tesla
  – GM
  – Toyota

• More than just vision
  – LIDAR
  – Planning
  – Mapping
  – Anticipating behavior of other drivers
Vision-based interaction (and games)

Microsoft Kinect
Virtual/Augmented Reality

- Tracking of user head with high accuracy
- Rendering realistic 3D scene in real-time
- Oculus / HTC / Hololens
Vision for robotics, space exploration

Vision systems (JPL) used for several tasks

- Panorama stitching
- 3D terrain modeling
- Obstacle detection, position tracking
- For more, read “Computer Vision on Mars” by Matthies et al.

Source: S. Seitz
The problem

• Want to make a computer understand images
• We know it is possible – we do it effortlessly!
The Human Eye

- Retina measures about $5 \times 5$ cm and contains $10^8$ sampling elements (rods and cones).
- The eye’s spatial resolution is about $0.01^\circ$ over a $150^\circ$ field of view (not evenly spaced, there is a fovea and a peripheral region).
- Intensity resolution is about 11 bits/element, spectral range is 400–700nm.
- Temporal resolution is about 100 ms (10 Hz).
- Two eyes give a data rate of about 3 GBytes/s!
• Vision is the most powerful of our own senses.

• Around 1/3 of our brain is devoted to processing the signals from our eyes.

• The visual cortex has around $O(10^{11})$ neurons.
Vision as data reduction

- Raw feed from camera/eyes:
  - $10^{7-9}$ Bytes/s

- Extraction of edges and salient features
  - $10^{3-4}$ Bytes/s

- High-level interpretation of scene
  - $10^{1-2}$ Bytes/s
Why don’t we just copy the human visual system?

- People try to but we don’t yet have a sufficient understanding of how our visual system works.

- $O(10^{11})$ neurons used in vision

- By contrast, latest CPUs have $O(10^8)$ transistors (most are cache memory)

- Very different architectures:
  - Brain is slow but parallel
  - Computer is fast but mainly serial

- Bird vs Airplane
  - Same underlying principles
  - Very different hardware
Admin Interlude
Course details

• Course webpage:
  – http://cs.nyu.edu/~fergus/teaching/vision

• Use Piazza:
  – https://piazza.com/class/jy91mtk94qeiy
  – Office hours:
    – Thursday, 6pm-7pm,
      i.e. right before class.
  – Room 514, 60 5th Ave (my office)
Textbooks

• Course does not use a textbook

• Deep Learning book (Goodfellow, Courville and Bengio)
  – http://www.deeplearningbook.org/

• Geometric vision:


What you need

• Access to a computer than can run PyTorch

• Open-source packages for Deep Learning
  – Google & download

• Assignments should be possible on decent laptop
  – Project may require GPU (depending)
Grading

• Assignments (55%) + Course project (45%)

• Assignments on the course webpage are outdated: new ones will appear

• 4 assignments in Torch (55% of total)
  • 0\textsuperscript{th} = 5\%, 1\textsuperscript{st} = 15\%, 2\textsuperscript{nd} = 10\%, 3\textsuperscript{rd} = 25\%
Graders & Tutors

• 2 graders [Priyank Pathak (pp1953@nyu.edu) and Riken Mehta (riken.mehta03@nyu.edu)]
• 2 tutors [Chandra Konkimalla (cpk290@nyu.edu) and Manikanta Yellapragada (msy290@nyu.edu)]
  • Office hours TBD
  • Will do recitations on PyTorch
Assignment 0

• Test of pre-requisites
  • Some linear algebra
  • Coding

• Due date is Thursday Sept 12\textsuperscript{th}
  • Before the drop-date for classes (Mon Sept 16\textsuperscript{th})

• If you find this assignment super difficult, this might not be the right class for you.
Course Project

• Competition around self-supervised learning
• Will provide PyTorch environment
• You need to come up with idea to try out
  – Come to office hours to discuss

• Work in pairs
• Submit report instead of final exam
Course Project (Backup Plan)

• Please choose by mid-October
• Will put ideas up on Piazza
• Feel free to come up with your own!
  – Come to office hours to discuss

• Work in pairs
  – Can use whatever platform you prefer

• Submit report instead of final exam
Syllabus

• Low-level vision
  – Edge, corner, feature detection
  – Stereo reconstruction
  – Structure from motion, optical flow

• High-level vision
  – Introduction to neural nets
  – Convolutional nets (ConvNets)
  – Object recognition
  – Face recognition
  – Video recognition

• Other topics
  – Image processing tasks
  – Recurrent nets (images + text)
  – Generative models
  – Unsupervised learning
What the course will NOT cover

• Biology relating to vision
  – Go to CNS

• Reading the latest papers
  – Not an advanced course
  – Although will cover recent work in recognition

• How to capture & enhance images
  – See Computational Photography course
Likely Deviations

• Dec 5th Class (NeurIPS 2019 Conference)

• May have guest lecturers give some classes:
  – My PhD students
  – Researchers from Facebook AI Research
End of Admin Interlude
Computer Vision:
A whole series of problems

- What is in the image?
  - Object recognition problem
- Where is it?
  - 3D spatial layout
  - Shape
- How is the camera moving?
- What is the action?
Image is a projection of world
An under-constrained problem
Stereo Vision

• By having two cameras, we can triangulate features in the left and right images to obtain depth.
• Need to match features between the two images:
  – Correspondence Problem
Geometry:
3D models of planar objects

[Fitzgibbon et. al]
[Zisserman et. al.]
Structure and Motion Estimation

Objective: given a set of images …

Want to compute where the camera is for each image and the 3D scene structure:

- Uncalibrated cameras
- Automatic estimation from images (no manual clicking)
Example

Image sequence

Camera path and points

[Fitzgibbon et. al]
[et. al. Zisserman]
Application: Augmented reality

original sequence
Augmentation
DynamicFusion

• Real-time 3D

• Video
Interpretation from limited cues
Shape from Shading

• Recover scene structure from shading in the image

• Typically need to assume:
  – Lambertian lighting, isotropic reflectance
Shape from Texture

- Texture provides a very strong cue for inferring surface orientation in a single image.
- Necessary to assume homogeneous or isotropic texture.
- Then, it is possible to infer the orientation of surfaces by analyzing how the texture statistics vary over the image.
Human motion detection
Johansson’s experiments [‘70s]
Can you tell what it is yet?
Object Recognition

• “Understand objects in image”

• Different tasks:

Classification:
Image contains bus (binary yes/no)

Detection:
Localize object instances
(bounding box or mask)

Semantic segmentation:
Label every pixel
Cameras & Image Formation

Slides from: F. Durand, S. Seitz, S. Lazebnik, S. Palmer
Overview

• The pinhole projection model
  – Qualitative properties
  – Perspective projection matrix

• Cameras with lenses
  – Depth of focus
  – Field of view
  – Lens aberrations

• Digital cameras
  – Types of sensors
  – Color
Let’s design a camera

- Idea 1: put a piece of film in front of an object
- Do we get a reasonable image?
• Add a barrier to block off most of the rays
  – This reduces blurring
  – The opening is known as the aperture
Pinhole camera model

• Pinhole model:
  – Captures pencil of rays – all rays through a single point
  – The point is called Center of Projection (focal point)
  – The image is formed on the Image Plane
Dimensionality Reduction Machine (3D to 2D)

3D world

2D image

What have we lost?

- Angles
- Distances (lengths)
Projection properties

• Many-to-one: any points along same visual ray map to same point in image

• Points → points
  – But projection of points on focal plane is undefined

• Lines → lines (collinearity is preserved)
  – But line through focal point (visual ray) projects to a point

• Planes → planes (or half-planes)
  – But plane through focal point projects to line
Vanishing points

• Each direction in space has its own vanishing point
  – All lines going in that direction converge at that point
  – Exception: directions parallel to the image plane

• All directions in the same plane have vanishing points on the same line
Perspective distortion

• Problem for architectural photography: converging verticals
Perspective distortion

• The exterior columns appear bigger
• The distortion is not due to lens flaws
Perspective distortion: People
Modeling projection

• The coordinate system
  – The optical center \((O)\) (aka focal point / center of projection) is at the origin
  – Optical axis is in \(z\) direction
  – The image plane is parallel to \(xy\)-plane (perpendicular to \(z\) axis)

Source: J. Ponce, S. Seitz
Modeling projection

- Projection equations
  - Compute intersection with image plane of ray from \( P = (x,y,z) \) to \( O \)
  - Derived using similar triangles
    \[
    (x, y, z) \rightarrow (f \frac{x}{z}, f \frac{y}{z}, f)
    \]
  - We get the projection by throwing out the last coordinate:
    \[
    (x, y, z) \rightarrow (f \frac{x}{z}, f \frac{y}{z})
    \]

Source: J. Ponce, S. Seitz
Homogeneous coordinates

\[(x, y, z) \rightarrow \left( f \frac{x}{z}, f \frac{y}{z} \right) \]

• Is this a linear transformation?
  • no—division by \( z \) is nonlinear

Trick: add one more coordinate:

\[(x, y) \Rightarrow \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad \text{homogeneous image coordinates} \]

\[(x, y, z) \Rightarrow \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad \text{homogeneous scene coordinates} \]

Converting \textit{from} homogeneous coordinates

\[
\begin{bmatrix} x \\ y \\ w \end{bmatrix} \Rightarrow \left( \frac{x}{w}, \frac{y}{w} \right) \quad \begin{bmatrix} x \\ y \\ z \\ w \end{bmatrix} \Rightarrow \left( \frac{x}{w}, \frac{y}{w}, \frac{z}{w} \right)
\]
Perspective Projection Matrix

- Projection is a matrix multiplication using homogeneous coordinates:

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1/f & 0 \\ 0 & 0 & 1/f & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \Rightarrow (f \frac{x}{z}, f \frac{y}{z})$$

divide by the third coordinate
Perspective Projection Matrix

- Projection is a matrix multiplication using homogeneous coordinates:

\[
\begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1/f & 0 \\
0 & 0 & 1/f & 1
\end{bmatrix}
\begin{bmatrix}
x \\
y \\
z \\
1
\end{bmatrix} =
\begin{bmatrix}
x \\
y \\
z/f \\
1
\end{bmatrix} \Rightarrow (f \frac{x}{z}, f \frac{y}{z})
\]

divide by the third coordinate

In practice: split into lots of different coordinate transformations…

\[
\begin{bmatrix}
2D \\
point (3x1)
\end{bmatrix} =
\begin{bmatrix}
\text{Camera to} \\
pixel coord. \\
\text{trans. matrix (3x3)}
\end{bmatrix} \begin{bmatrix}
\text{Perspective} \\
projection matrix (3x4)
\end{bmatrix} \begin{bmatrix}
\text{World to} \\
camera coord. \\
\text{trans. matrix (4x4)}
\end{bmatrix} \begin{bmatrix}
\text{3D} \\
point (4x1)
\end{bmatrix}
\]
Orthographic Projection

- Special case of perspective projection
  - Distance from center of projection to image plane is infinite

- Also called “parallel projection”
- What’s the projection matrix?
Building a real camera
Camera Obscura

- Basic principle known to Mozi (470-390 BCE), Aristotle (384-322 BCE)

- Drawing aid for artists: described by Leonardo da Vinci (1452-1519)

Source: A. Efros
Home-made pinhole camera

Why so blurry?

http://www.debevec.org/Pinhole/
Shrinking the aperture

- Why not make the aperture as small as possible?
  - Less light gets through
  - Diffraction effects…
Shrinking the aperture
Adding a lens

• A lens focuses light onto the film
  – Rays passing through the center are not deviated
Adding a lens

- A lens focuses light onto the film
  - Rays passing through the center are not deviated
  - All parallel rays converge to one point on a plane located at the focal length $f$
Adding a lens

- A lens focuses light onto the film
  - There is a specific distance at which objects are “in focus”
    - other points project to a “circle of confusion” in the image
Thin lens formula

\[ f = \frac{D \cdot D'}{D - D'} \]
Thin lens formula

Similar triangles everywhere!

\[ D' \quad D \quad f \]
Thin lens formula

\[ \frac{y'}{y} = \frac{D'}{D} \]
Thin lens formula

\[
y' / y = D' / D \\
y' / y = (D' - f) / f
\]
Thin lens formula

\[ \frac{1}{D'} + \frac{1}{D} = \frac{1}{f} \]

Any point satisfying the thin lens equation is in focus.
Depth of Field

http://www.cambridgeincolour.com/tutorials/depth-of-field.htm

Slide by A. Efros
How can we control the depth of field?

- Changing the aperture size affects depth of field
  - A smaller aperture increases the range in which the object is approximately in focus
  - But small aperture reduces amount of light – need to increase exposure
Varying the aperture

Large aperture = small DOF

Small aperture = large DOF
Field of View

From London and Upton
Field of View

From London and Upton
Field of View

$\varphi = \tan^{-1}\left(\frac{d}{2f}\right)$

Smaller FOV = larger Focal Length
Field of View / Focal Length

Large FOV, small f
Camera close to car

Small FOV, large f
Camera far from the car

Sources: A. Efros, F. Durand
Same effect for faces

wide-angle  standard  telephoto

Source: F. Durand
Approximating an affine camera

Source: Hartley & Zisserman
Real lenses
Lens Flaws: Chromatic Aberration

- Lens has different refractive indices for different wavelengths: causes color fringing

Near Lens Center
Near Lens Outer Edge
Lens flaws: Spherical aberration

- Spherical lenses don’t focus light perfectly
- Rays farther from the optical axis focus closer
Lens flaws: Vignetting
Radial Distortion

- Caused by imperfect lenses
- Deviations are most noticeable near the edge of the lens
Digital camera

- A digital camera replaces film with a sensor array
  - Each cell in the array is light-sensitive diode that converts photons to electrons
  - Two common types
    - Charge Coupled Device (CCD)
    - Complementary metal oxide semiconductor (CMOS)
CCD vs. CMOS

- **CCD**: transports the charge across the chip and reads it at one corner of the array. An *analog-to-digital converter (ADC)* then turns each pixel's value into a digital value by measuring the amount of charge at each photosite and converting that measurement to binary form.

- **CMOS**: uses several transistors at each pixel to amplify and move the charge using more traditional wires. The CMOS signal is digital, so it needs no ADC.


Color sensing in camera: Color filter array

Bayer grid

Estimate missing components from neighboring values (demosaicing)

Why more green?

Human Luminance Sensitivity Function

Source: Steve Seitz
Problem with demosaicing: color moire
The cause of color moire

Fine black and white detail in image misinterpreted as color information
Color sensing in camera: Foveon X3

- CMOS sensor
- Takes advantage of the fact that red, blue and green light penetrate silicon to different depths


better image quality

Source: M. Pollefeys
Digital camera artifacts

• **Noise**
  - low light is where you most notice noise
  - light sensitivity (ISO) / noise tradeoff
  - stuck pixels

• **In-camera processing**
  - oversharpening can produce halos

• **Compression**
  - JPEG artifacts, blocking

• **Blooming**
  - charge overflowing into neighboring pixels

• **Color artifacts**
  - purple fringing from microlenses,
  - white balance
Historic milestones

- **Pinhole model**: Mozi (470-390 BCE), Aristotle (384-322 BCE)
- **Principles of optics (including lenses)**: Alhacen (965-1039 CE)
- **Camera obscura**: Leonardo da Vinci (1452-1519), Johann Zahn (1631-1707)
- **First photo**: Joseph Nicephore Niepce (1822)
- **Daguerréotypes** (1839)
- **Photographic film** (Eastman, 1889)
- **Cinema** (Lumière Brothers, 1895)
- **Color Photography** (Lumière Brothers, 1908)
- **Television** (Baird, Farnsworth, Zworykin, 1920s)
- **First consumer camera with CCD**: Sony Mavica (1981)
- **First fully digital camera**: Kodak DCS100 (1990)