The University of Texas at Austin
Electrical and Computer Engineering
Cockrell School of Engineering

Spring 2021

ADVANCED TOPICS IN COMPUTER VISION

Atlas Wang
Assistant Professor, The University of Texas at Austin

Visual Informatics Group@UT Austin
https://vita-group.github.io/

Course Logistics

- We meet on Zoom (link sent), 5 – 6:30pm, M&W
  - In class, you can submit your questions on Zoom Chat
- Class materials are distributed on **Course Webpage**: https://vita-group.github.io/spring_21.html
- After-class communication: **Slack (link sent)**
- Office Hour: no regular & by appointment
- No TA for this class (i.e., I'm your "super TA")
- Can I audit or sit in?

Grading

- **Class Participation: 10%**
- **Three In-Class Quizzes: 10% each**
  - 10-15 minutes: time TBD; will notify ahead of time
- **Final Project: 60%**
  - Proposal (**15%)** Due by the end of **Week 4**: 2-Page report, including project title, team member, problem description, preliminary literature survey, the proposed technical plan, and references
  - Presentation (**15%**): Be prepared to be challenged by your peers and the instructor
  - Code review (**15%**): Write clean, well-documented and runnable codes, PLEASE
  - Final Report (**15%**): 8+1 page report following the standard CVPR paper template (and quality level)
    - Template: http://cvpr2020.thecvf.com/sites/default/files/2019-09/cvpr2020AuthorKit.zip

Project Guidance

- **Teaming:** we encourage 2 students to form a team, as you are expected to carry on a semester-long research project with substantial innovations.
  - A small number of single-person teams may be approved by the instructor too, if well justified.
- Each project team has to be registered to and approved by the end of **Week 2**.
  - A Google Sheet will be provided for team registration
- **Topic:** your choice, but must be relevant to computer vision
  - What if I don't have a specific idea now ? Talk to the instructor on Slack …
- **Extra credits** will be given to:
  - One project to receive the Best Project Award, *voted by all class members* (+5%)
  - Projects in **interdisciplinary domains** (some examples: 5G/6G telecommunication, brain-computer interface, economics & markets, COVID-19, etc.), *judged by the instructor* (+2%)

How to Develop Good Project Timeline

- There is no weekly or intermediate report to me, but mark your own timeline
- **First things first:** conduct a thorough literature survey to avoid reinventing wheels, and then discuss with the instructor
- Don’t delay yourself until last minute. The project should be scheduled as **one full semester long**: it is NOT something that you can rush in a day or two!
- Discuss and divide task assignments with your teammate. **Everyone needs to perform** (and who did what needs to be explicitly discussed in the report)

How to Write Good Proposal & Report

- What's the problem definition? Why it is important? What were done in literature (try summarizing & categorizing)? What remain to be the main challenges? What technical gap do you aim to reduce? **[TBD in your proposal]**

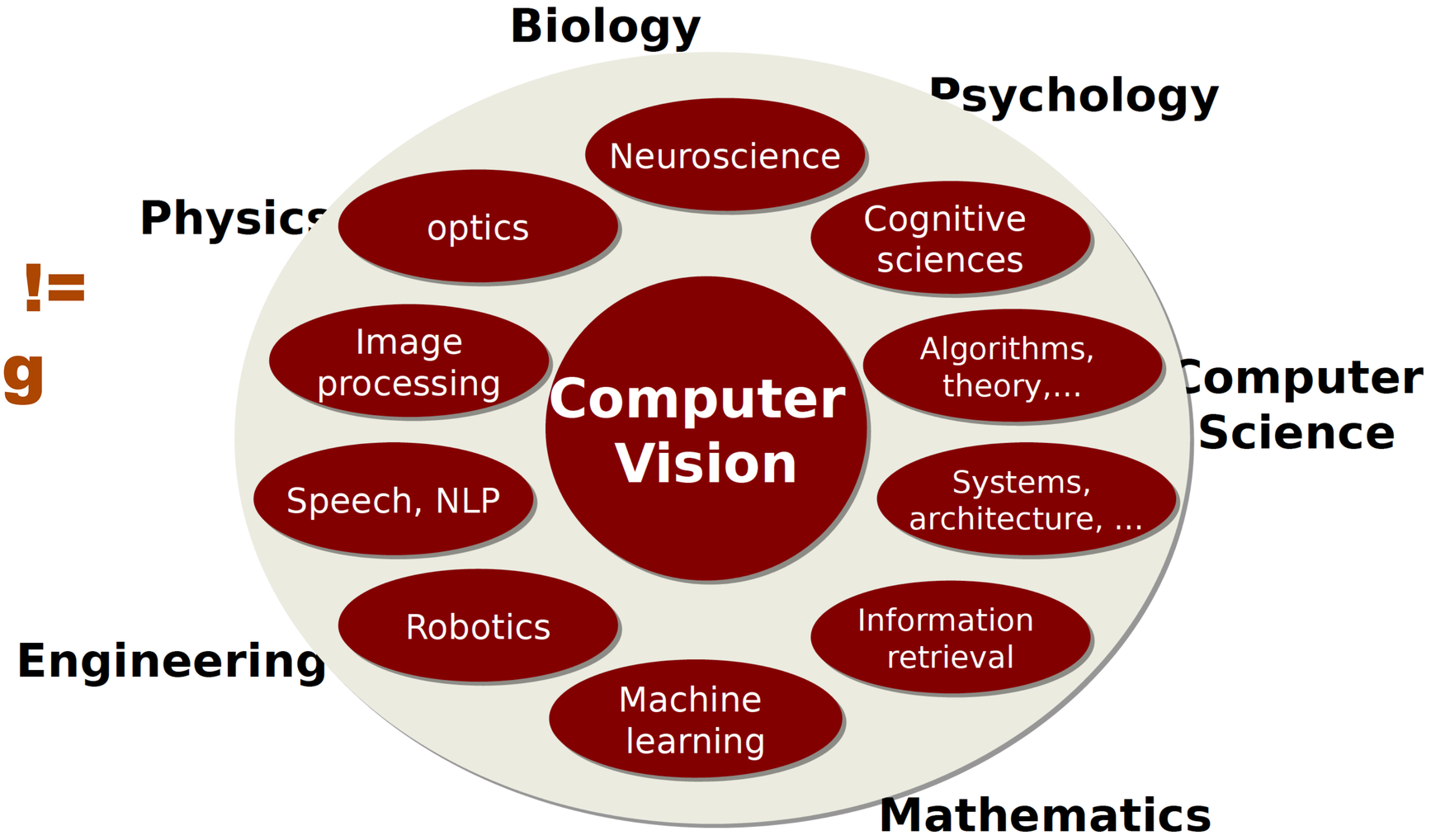
- What are the experiment settings? What are the main baselines to compare with? What are the main advantages and drawbacks of your idea as shown by experiments? What are potential future works? **[TBD in your report]**

- It's not easy to fill a CVPR template.
  - FYI: If I devote full energy to writing a CVPR draft from scratch (with all technical work already done), it'll take me ~2 full days
  - **Use Latex, Use Latex, Use Latex. Word not accepted!!**

Overview & Prerequisite

- Computer vision is a HUGE field. This class is designed to cover just “several drops” in the ocean (biased towards the “hot and fresh” frontiers)
- Lectures are mixture of detailed techniques and high-level ideas.
- Computer vision is a highly technical field: know your math & be a good coder!
- This class is NOT designed for pure “beginners”. We will speak technical language from Day 1.
  - You are assumed to already be familiar with: Linear Algebra, Convex Optimization, Probability & Stochastic Process
  - You are assumed to know the basics about (but not an expert on): Digital Signal Processing, Image & Video Processing, Machine Learning & Data Mining

Computer Vision != Machine Learning



What is Computer Vision?

- An interdisciplinary field that deals with how computers can be made for gaining holistic understanding from digital images or videos.
- From the engineering perspective, it seeks to automate tasks that the human visual system can do.

Computer Vision as Input-Output System:

- Input: images or video
- Output (ideally): description or understanding of the visual world, in a “human” way
- Outputs (practically): reconstructing, measuring, classifying, interpreting…

Do you know?

The first “Computer Vision” work in this world was originally a summer project given to an MIT undergraduate student

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

PROJECT MAC

Artificial Intelligence Group — July 7, 1966

Vision Memo. No. 100.

THE SUMMER VISION PROJECT

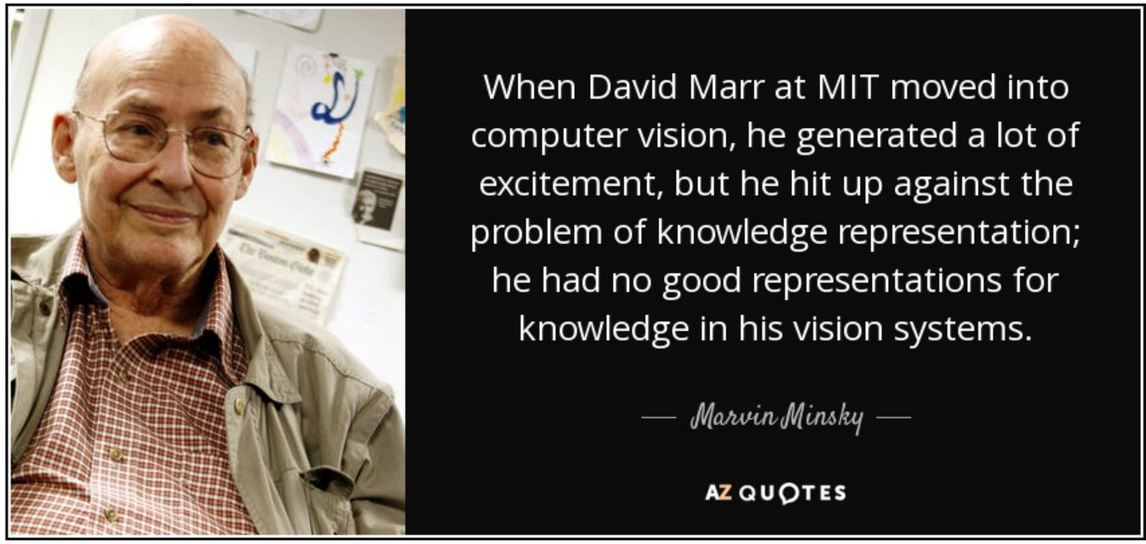
Seymour Papert

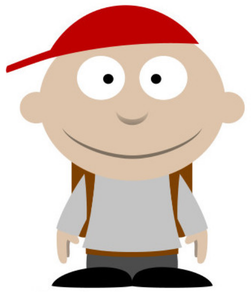
The summer vision project is an attempt to use our summer workers effectively in the construction of a significant part of a visual system. The particular task was chosen partly because it can be segmented into sub-problems which will allow individuals to work independently and yet participate in the construction of a system complex enough to be a real landmark in the development of "pattern recognition".

Computer Vision Research History: My (probably approximately correct) summary

- **Late 1960s**: CV was born = a branch of human vision and cognition research (*bio-inspired CV*)
- **1970s**: CV = estimate 3D structures from 2D images (*physically-grounded CV*)
- **1980s**: more rigorous math concepts such as scale space, texture analysis, contour models, as well as the emergence of optimization and inference methods
- **Early-to-mid 1990s:** camera calibration, multi-view stereo, scene reconstruction, image segmentation, the big boom of statistical learning methods
- **Late 1990s:** bridging CV and graphics: rendering, morphing, stitching…
- **2000s and after**: ML (graphical models, sparsity & low-rank), and finally Deep Learning ...

After 55 Years…Computer Vision is Still Tough!





Atlas Wang

Hey Tom, What do you see as the biggest problem in computer vision?

One biggest problem of computer vision is – human never see in pixels!

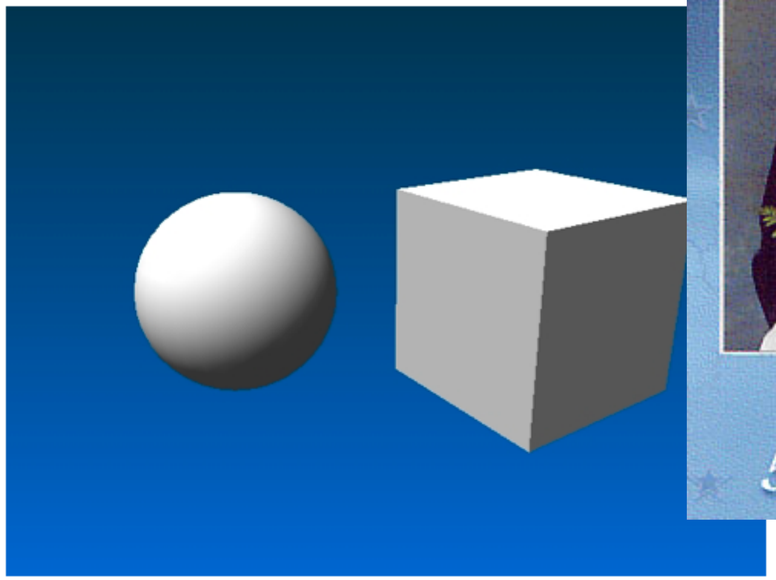
Prof. Thomas S. Huang (1936 - 2020), ECE@UIUC
"A founding father in computer vision"

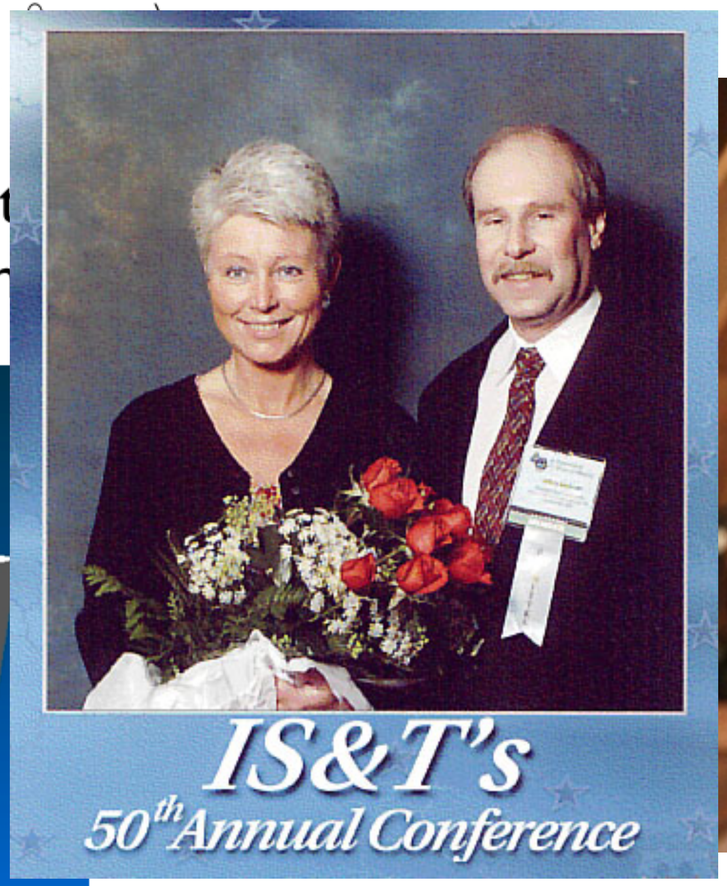
- Situation much the same as AI:
  - Some fundamental algorithms
  - Large collection of hacks / heuristics
- CV research is hard and "never eneding"
  - Especially at high level, physiology unknown
  - Requires integrating many different methods
  - Requires reasoning and understanding: "AI completeness"

(cube, size, x_0, y_0, z_0, θ_{XY}, θ_{XZ}, ...

(sphere, radius, x_1, y_1, z_1, ...

Computer Graphics ↓ ↑ Comput Vision



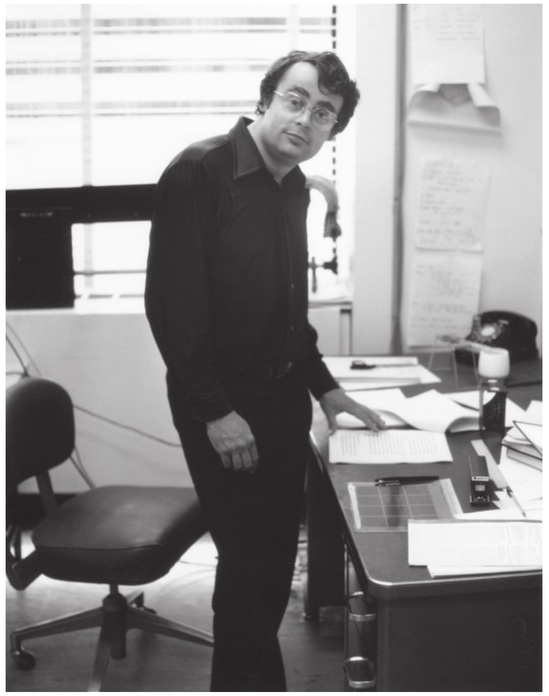


Computer Vision and Computer Graphics are often viewed as “inverse operations”

Computer Vision and Image Processing are significantly overlapped in their tools

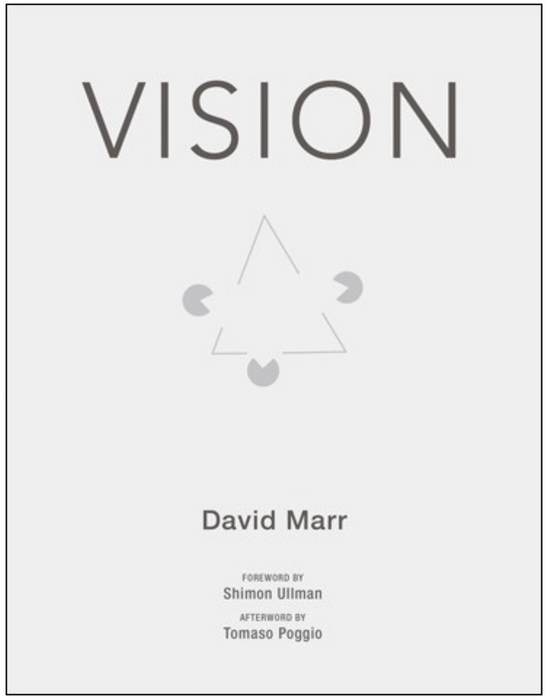
(http://www.cs.cmu.edu/~chuck/lennapg/lenna.shtml)

Marr's Tri-Level Hypothesis for Vision



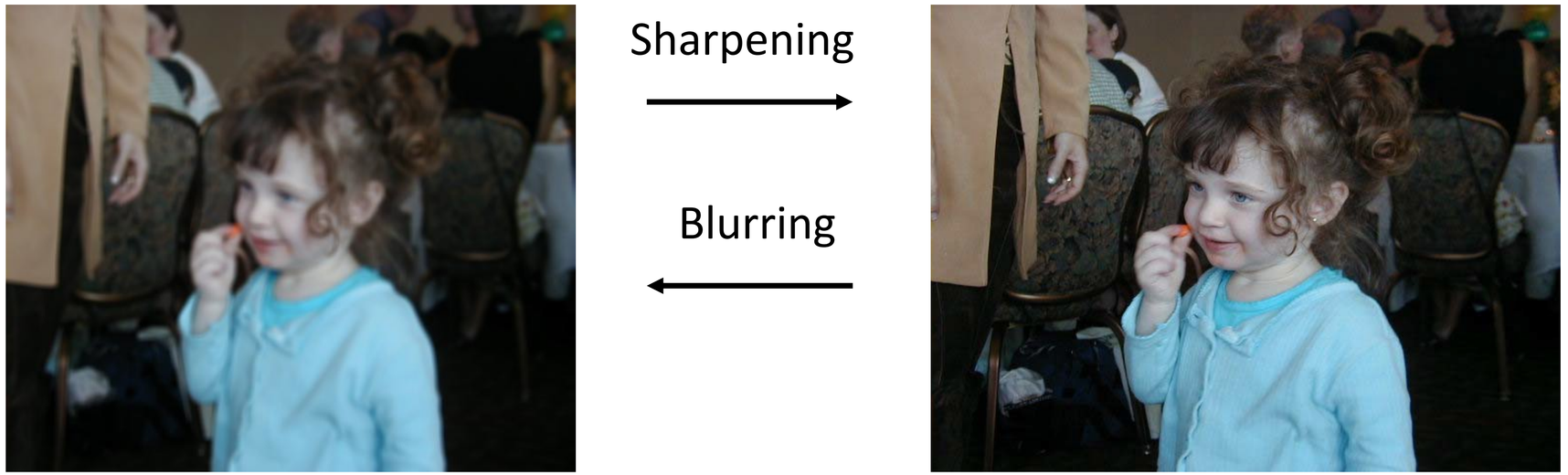
David Marr integrated results from psychology, artificial intelligence, and neurophysiology into new models of visual processing, creating the field of Computer Vision.

- **Computational Level:** what does the system do (e.g.: what problems does it solve or overcome) and similarly, why does it do these things -- What is the problem?
- **Algorithmic level** (a.k.a. **representational level**): how does the system do what it does, specifically, what representations does it use and what processes does it employ to build and manipulate the representations -- How to solve the problem?
- **Implementational level** (a.k.a. **physics level**): how is the system physically realized (in the case of biological vision, what neural structures and neuronal activities implement the visual system) -- How the above are done in a computer or a brain?



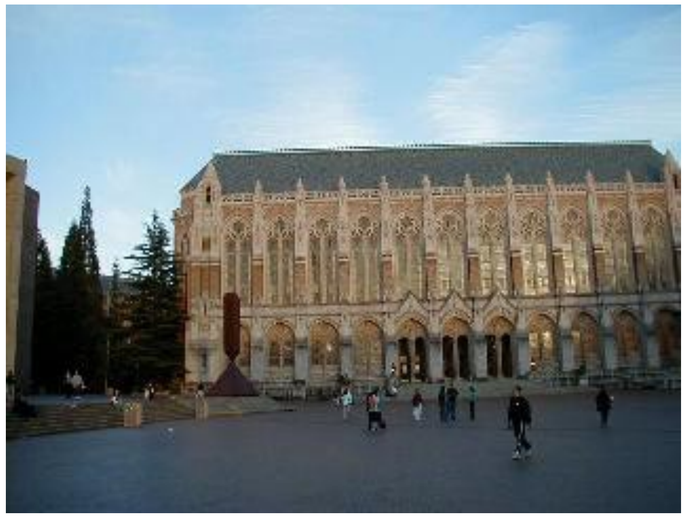
Three Stages in Computer Vision

- **Low-Level:** Image to image (enhancement, edge detection…)
  - Largely overlapped with signal or image “reconstruction” & “filtering”
  - Directly interface with image formulation, often considered as “pre-processing” for CV tasks

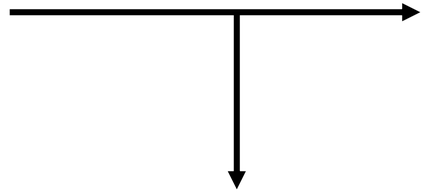


Three Stages in Computer Vision

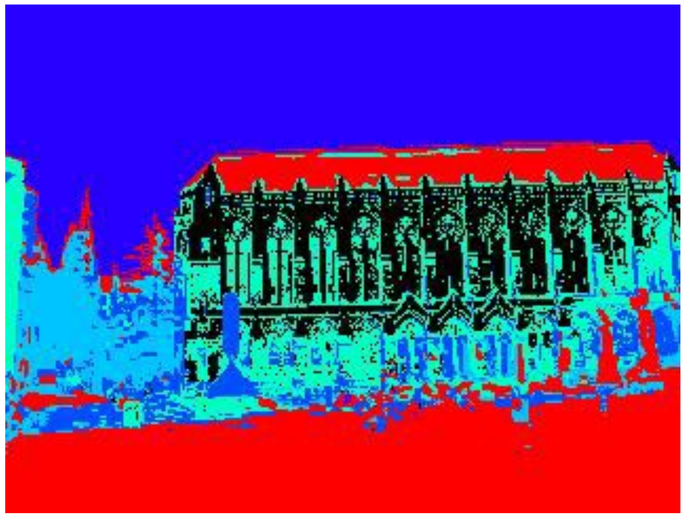
- **Mid-Level:** Image to feature (classical segmentation, grouping…)
  - ***What's the criterion?*** Gestalt psychologists suggest an intermediate vision stage whose underlying processes are *grouping* mechanisms, which are essential for separating objects from background. Certain "*commonsense*" principles, such as closure, symmetry, or similarity guide how to group pieces of image and locate boundary.



Clustering + connected component analysis

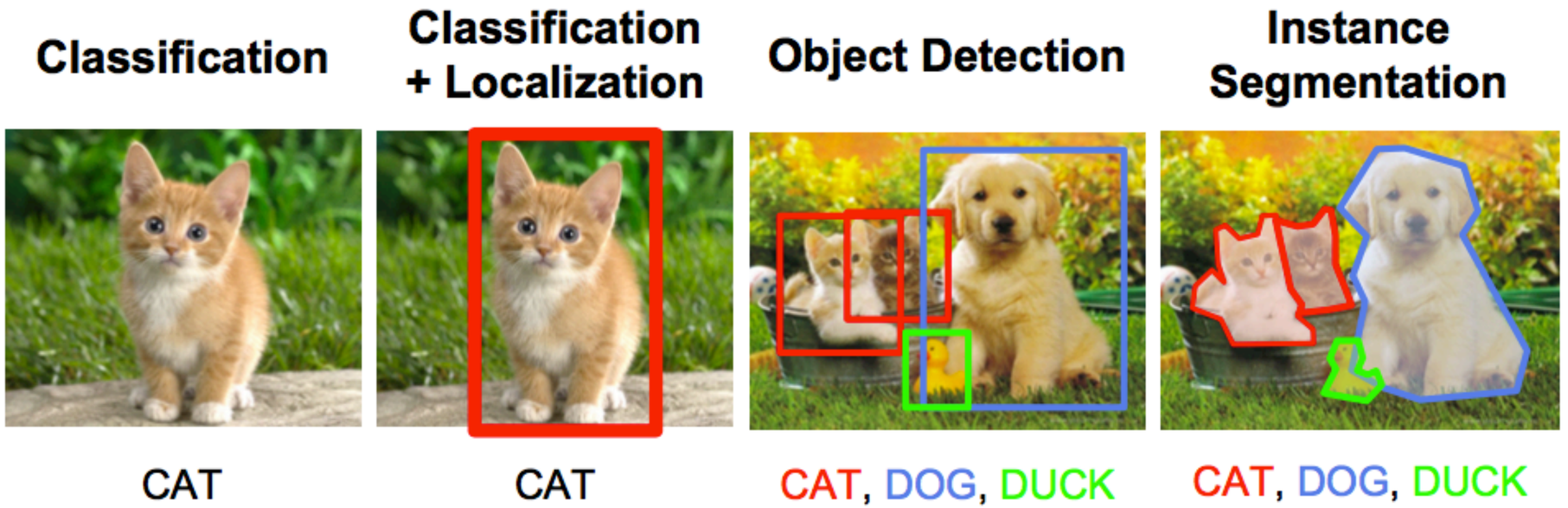


Object Structure



Three Stages in Computer Vision

- **High-Level:** Image to analysis (recognition, detection, semantic segmentation …)
  - Facilitating semantic interpretation of visual data, and required for numerous applications like robotics, driver assistance, multi-media retrieval, biometrics and surveillance …



Three Levels: An Example

“There’s an edge!”

“There’s an object and a background!”

“There’s a chair!”

Example: A Simple Computer Vision Pipeline (1990s)

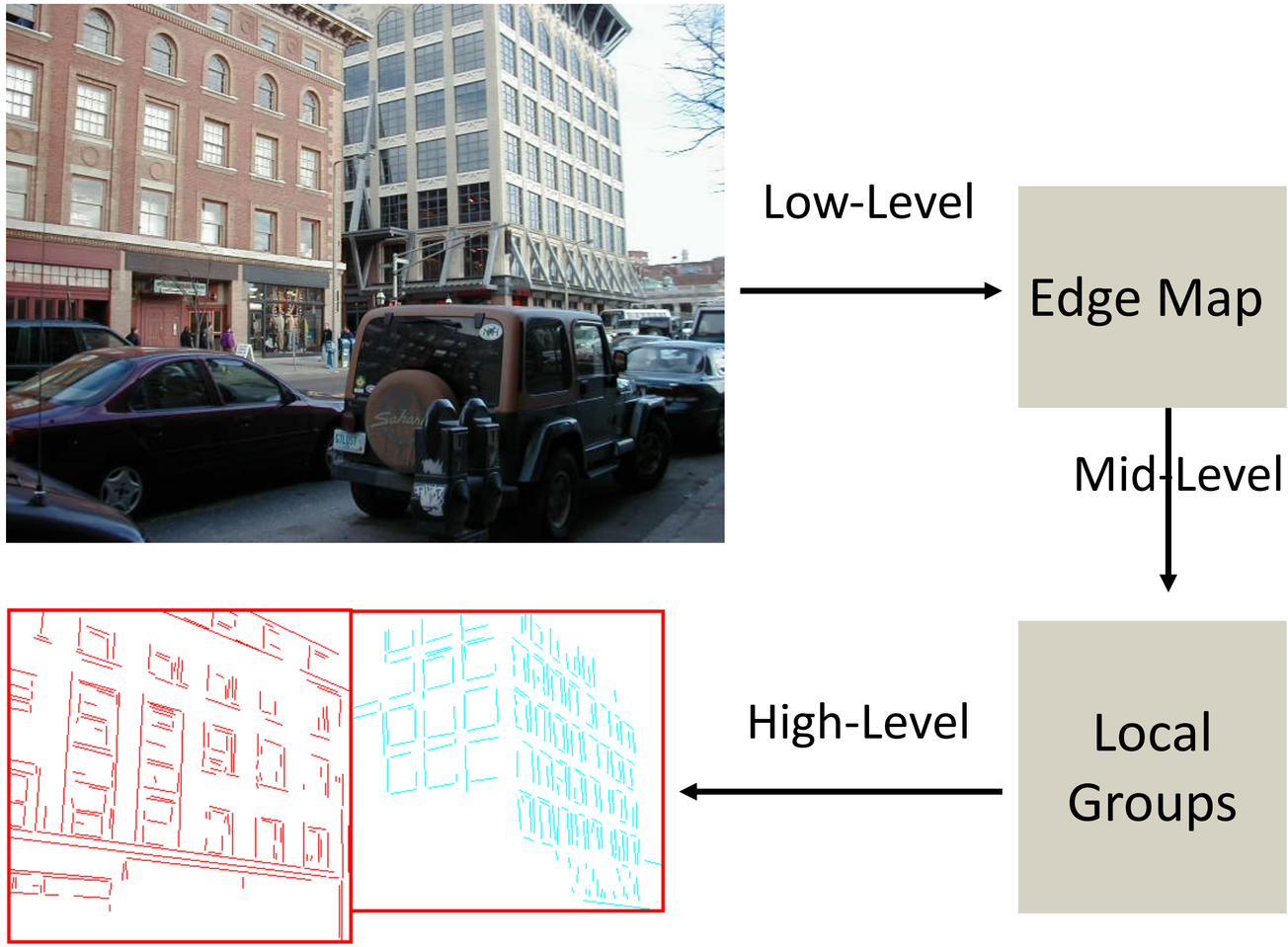
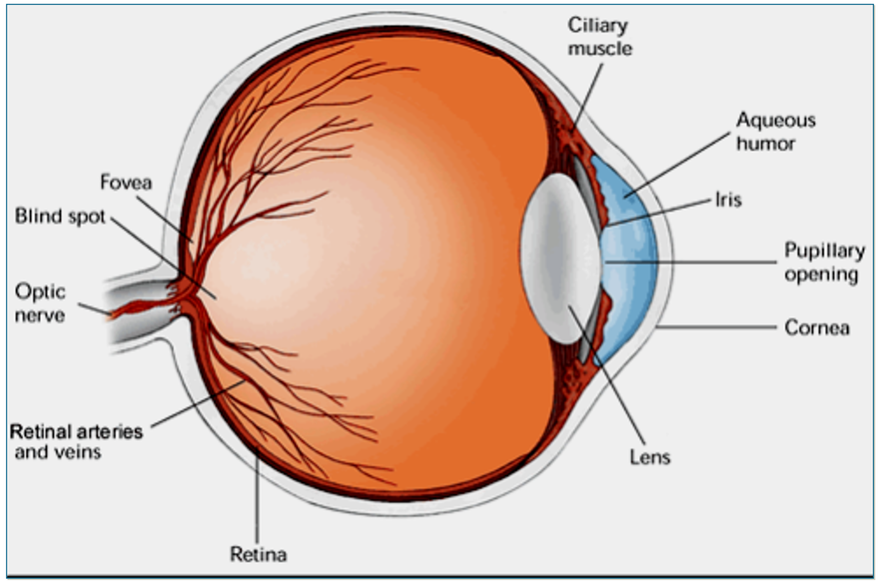
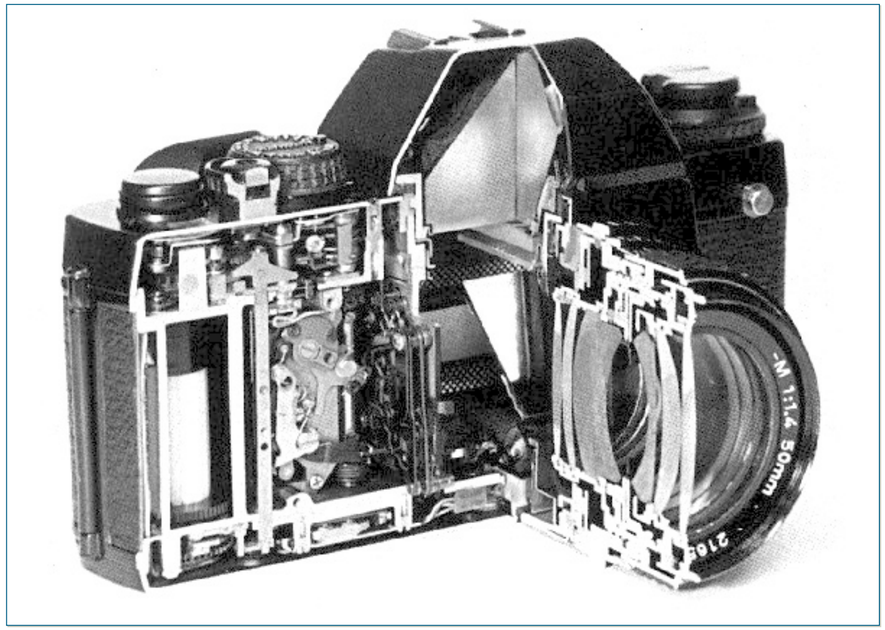


Image Formulation

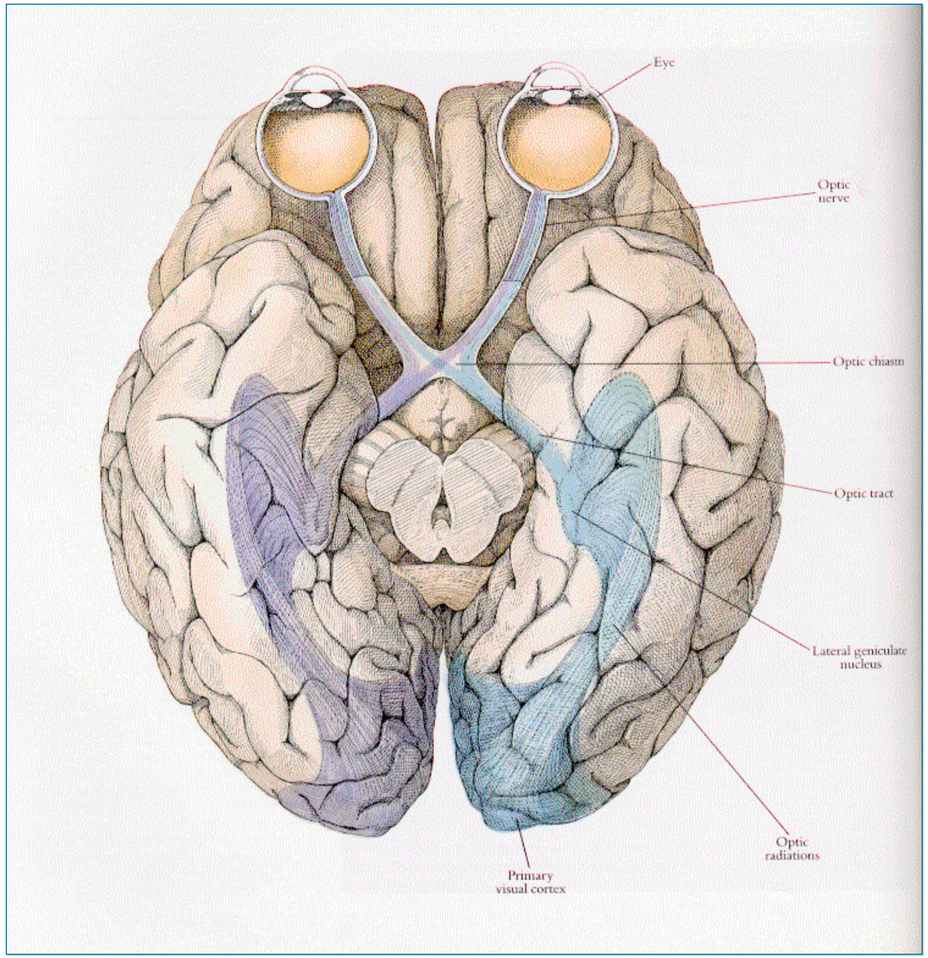
- Belong to “low-level” vision
- Human: lens forms image on retina, sensors (rods and cones) respond to light
- Computer: lens system forms image, sensors (CCD, CMOS) respond to light





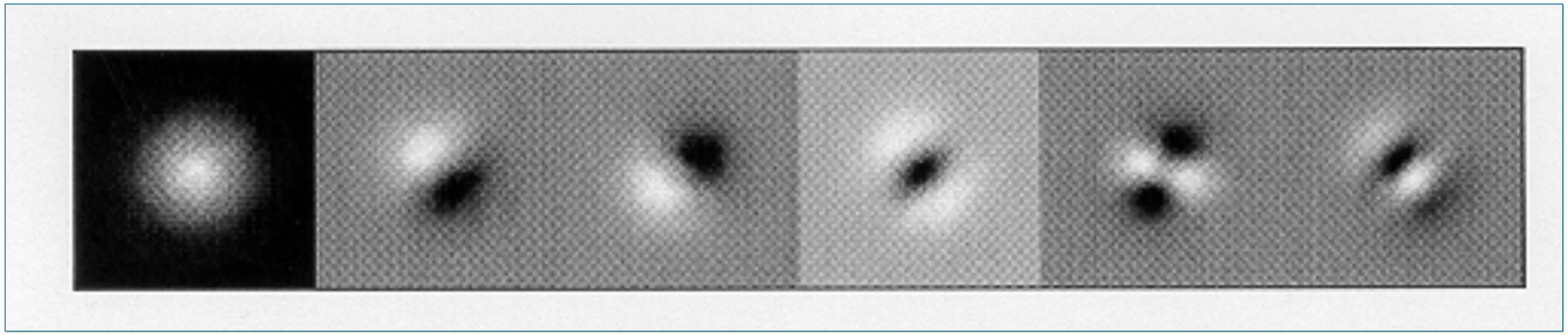
Low-Level Human Vision

- Retinal ganglion cells
- Lateral Geniculate Nucleus – function unknown (visual adaptation?)
- Primary Visual Cortex (“**magnitude and phase**”)
  - Simple cells: orientational sensitivity
  - Complex cells: directional sensitivity
- Further processing (“**what-where pathway**”)
  - Temporal cortex: what is the object?
  - Parietal cortex: where is the object? How do I get it?



Low-Level Human Vision: Spatial

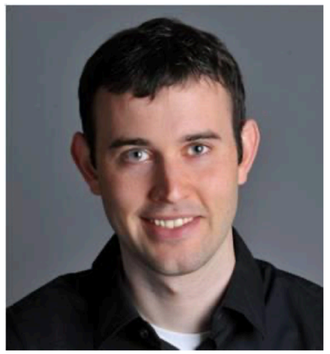
- "Net effect": low-level human vision can be (partially) modeled as a set of *multi-resolution, and multi-orientation* filters



- Human perception cues are dominated by mid- to high-frequency bands
  - When we see something from a distance, we are effectively subsampling it ()
- "Depth Cues": Focus, Vergence, Stereo …

Example: Hybrid Images

?

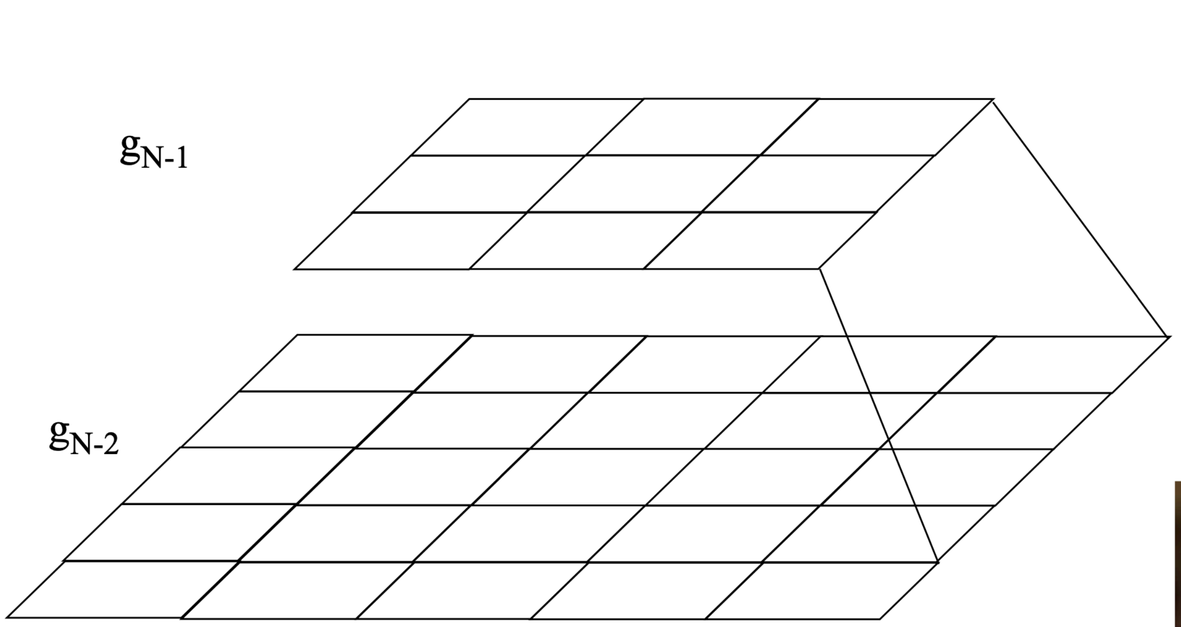


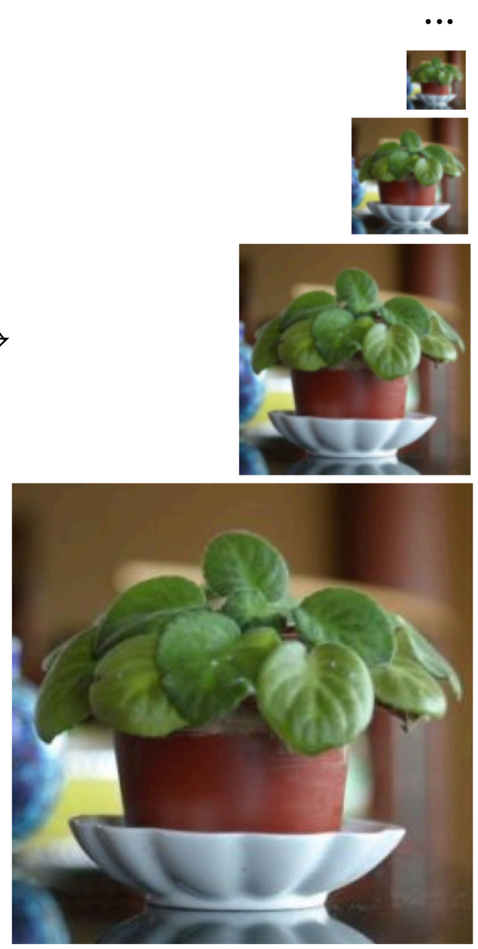
- Distance-dependent perception of hybrid images by human
- *Are you still complaining deep networks are easily fooled?*☺

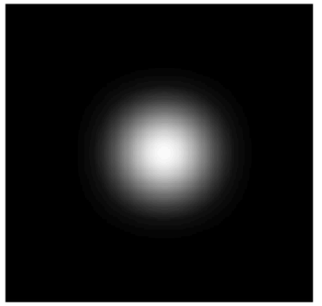
Low-Level Computer Vision: Spatial

- **Critical Building Block:** Filters and filter banks
  - Often implemented via **convolution (yes, that guy in deep learning)**
  - Detection of edges, corners, and other local features
  - Texture pattern recognition and synthesis
  - Can include multiple orientations
  - Can include **multiple scales:** “filter pyramids” – a key feature in deep learning

Example: Gaussian Pyramid

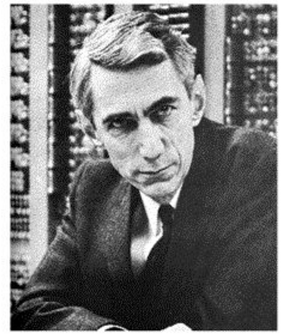




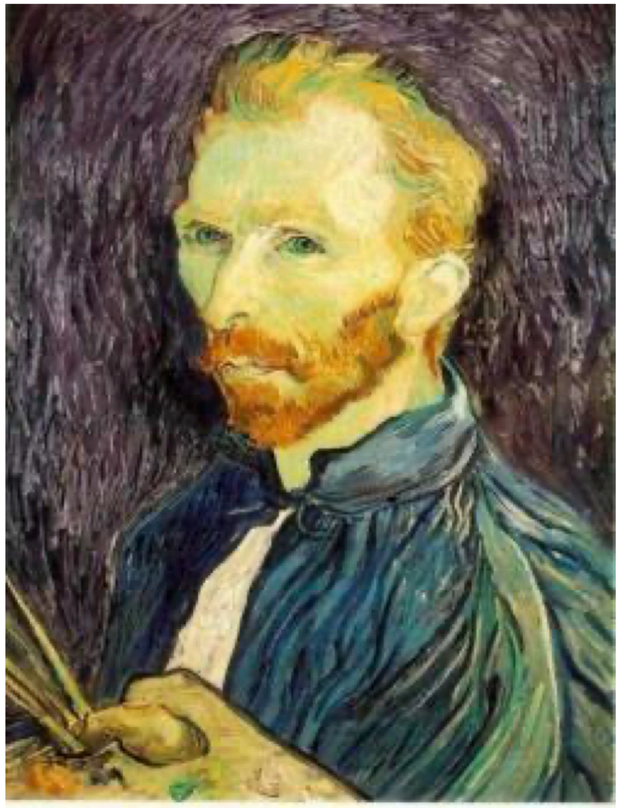


g_0 is the original image

Why not just naively sub-sampling?


Claude Shannon


Harry Nyquist

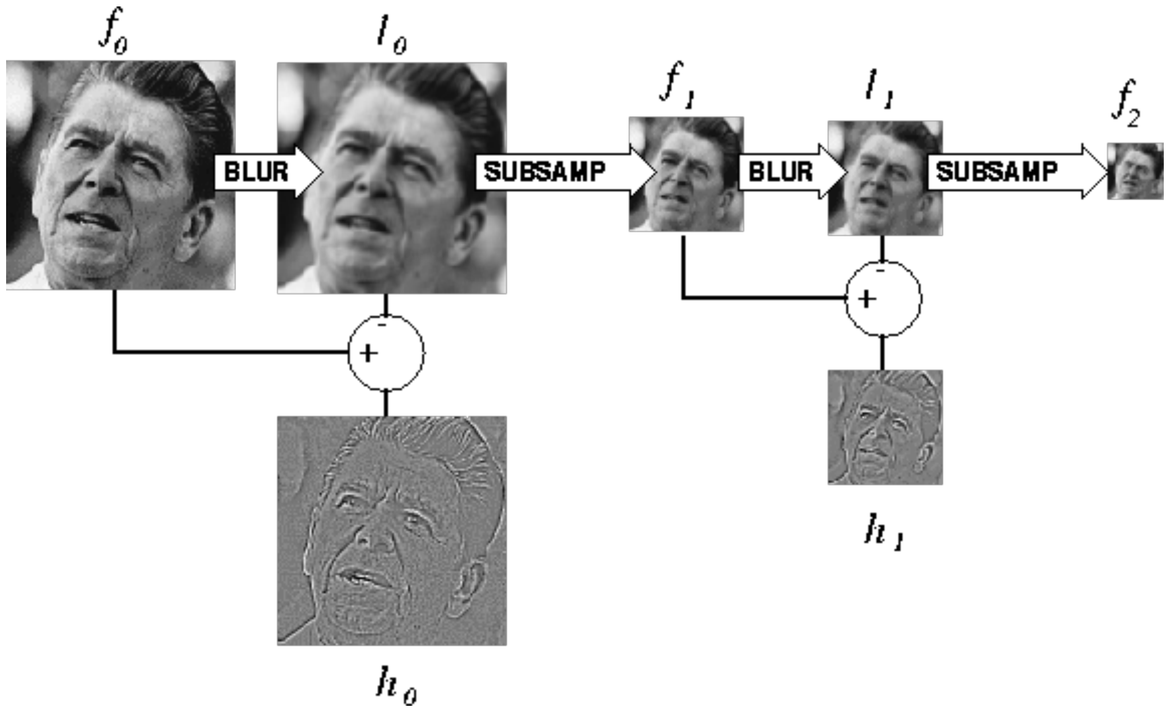

Gaussian 1/2

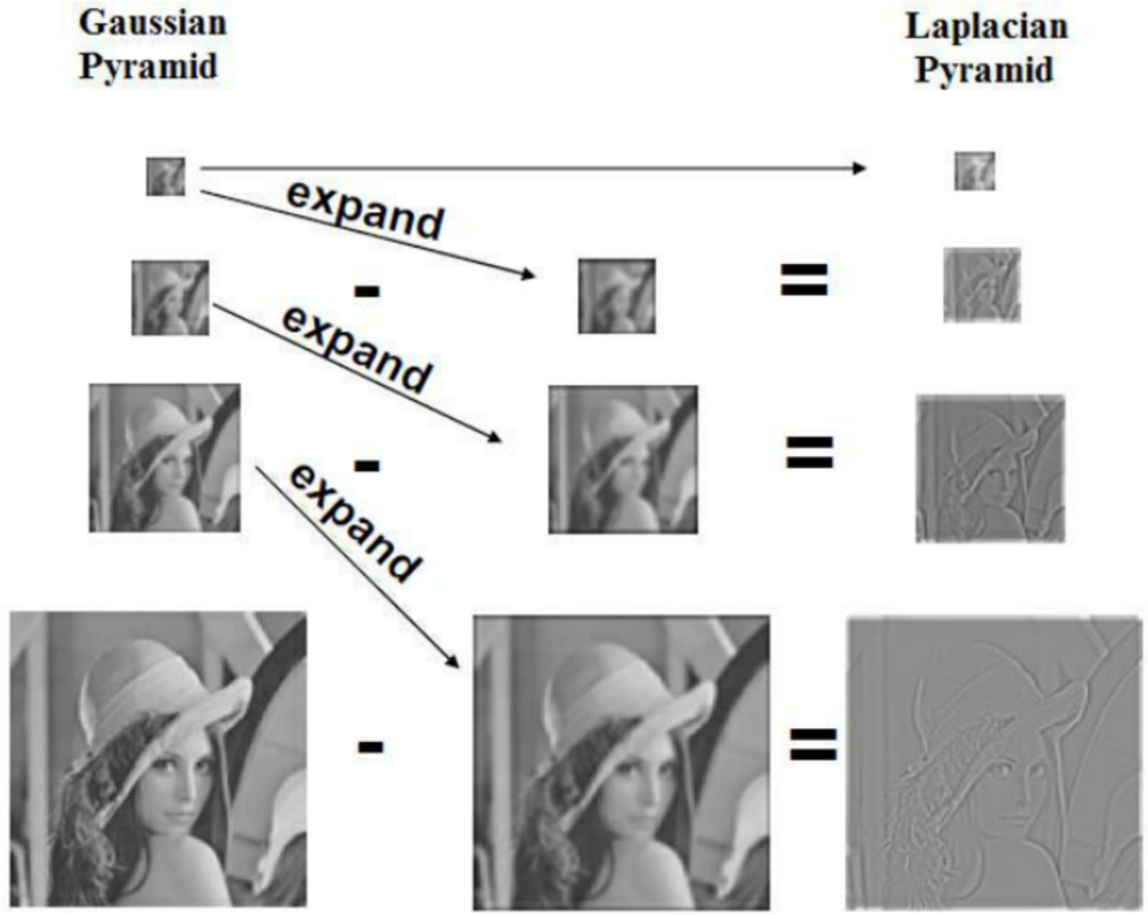

G 1/4


G 1/8

- **Aliasing** occurs when your sampling rate is not high enough to capture high-frequency details
- Smoothening, e.g., by Gaussian, is to reduce the maximum frequency of image features
- First smoothening, then sub-sampling!

Example: Laplacian Pyramid





Now You Can Create A Hybrid Image!

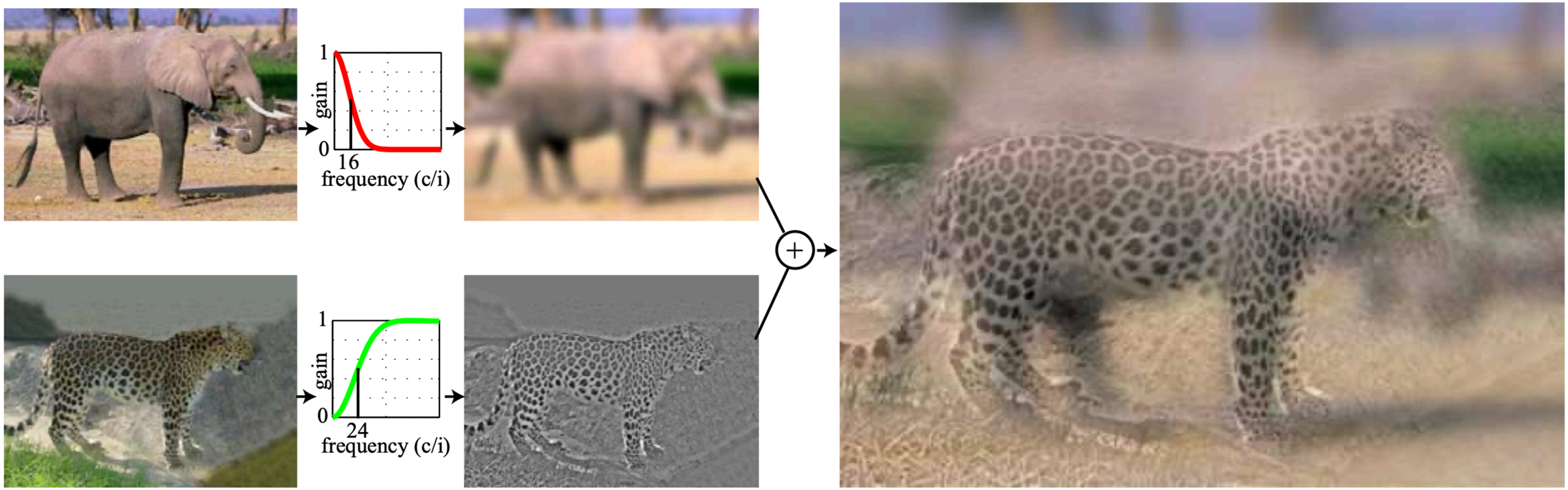
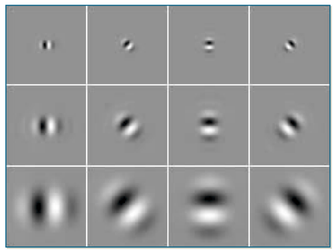
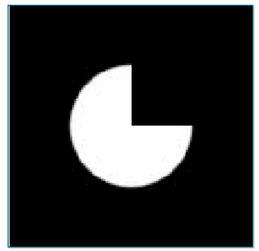


Figure 2: hybrid images are generated by superimposing two images at two different spatial scales: the low-spatial scale is obtained by filtering one image with a low-pass filter, and the high spatial scale is obtained by filtering a second image with a high-pass filter. The final hybrid image is composed by adding these two filtered images.

Example: Multi-Scale Texture Analysis



Multiresolution
Oriented Filter Bank



Original
Image

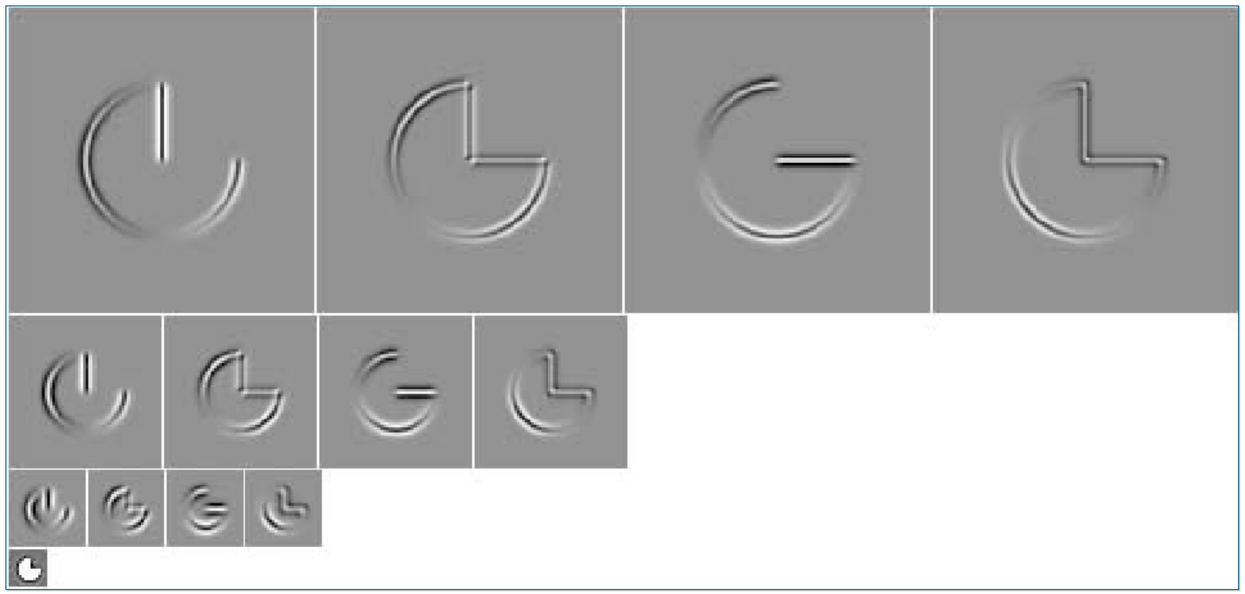


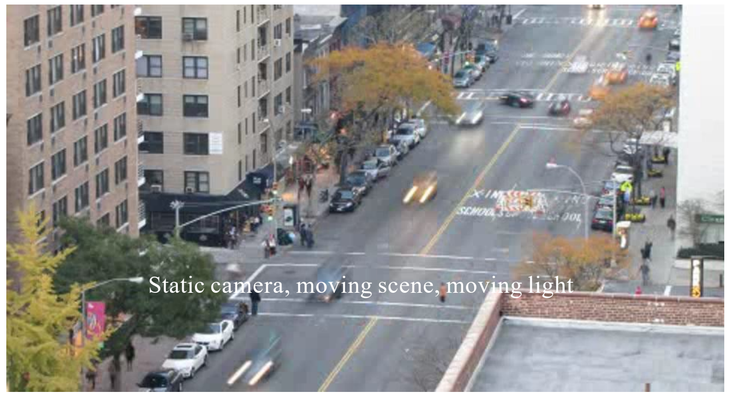
Image
Pyramid

Low-Level Vision: Temporal

Static camera, moving scene

Moving camera, static scene

Moving camera, moving scene



Static camera, moving scene, moving light

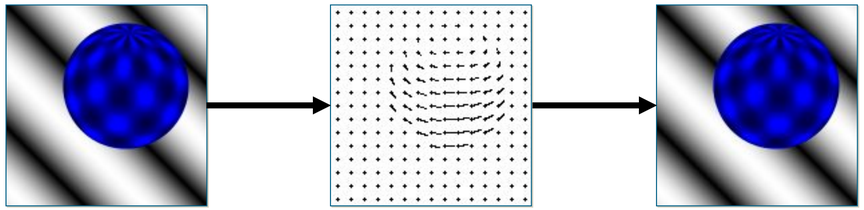
Motion in Computer Vision: Cause

- Three factors in imaging process
  - Light (time lapse)
  - Scene (object movement)
  - Camera (shake)
- Varying any of them can cause motion
- **More sophisticated natural motions exist …**

Motion in Computer Vision: Analysis

- Challenges of motion estimation
  - Geometry: shapeless objects
  - Reflectance: transparency, shadow, reflection
  - Lighting: fast moving light sources
  - Sensor: motion blur, noise

Example: Optical flow



- Key for analysis: motion representation
  - Ideally, solve the inverse rendering problem for a video sequence -- Intractable!
  - Practically, we make strong assumptions
    - Geometry: rigid or slow deforming objects
    - Reflectance: opaque, Lambertian surface
    - Lighting: fixed or slow changing
    - Sensor: no motion blur, low-noise

“Natural Image Manifold”

- The distribution of natural images (or patches) is similar to the mass distribution in the universe, where there are high-density and low-density areas
- This “image manifold” has to be highly nonlinear, inherently **low-dimensional**, and **locally smooth** … (why?)

The University of Texas at Austin

Electrical and Computer Engineering

Cockrell School of Engineering

Digital Image Representation

Binary

Gray scale

Color

Digital Images are Sampled and Quantized

- An image contains discrete number of pixels

  Pixel value:

  - "grayscale" (or "intensity"): [0,255]
  - "color"
    - –RGB: [R, G, B]

[90, 0, 53]

[249, 215, 203]

[213, 60, 67]

Image Sampling

Binary

Gray scale

Color