Computer Vision: Summary and Discussion

Computer Vision

CS 543 / ECE 549

University of Illinois

Derek Hoiem

HW 5 – PCA/FLD

- Why did training with subsets 1+5 (vs. subset 1 only) make PCA worse but FLD better?

S1

S5

Method (train set)	Subset 1	Subset 2	Subset 3	Subset 4	Subset 5
PCA (S1) (d=9/30)	0/0	0/0	0.225/0.042	0.664/0.564	0.858/0.774
FLD (S1) (c=10/31)	0/0	0/0	0.025/0.025	0.457/0.457	0.874/0.874
PCA (S1+S5) (d=9/30)	0/0	0.167/0	0.725/0.342	0.693/0.289	0/0
FLD (S1+S5) (c=10/31)	0/0	0/0	0/0	0.014/0.028	0/0

HW 5 – Deep Nets

Edward Xue (0.654)

- Network-in-network
- Careful learning rate tuning

- **Network Architecture**

Layer	type	Input	Filter	stride	pad	Output
1	conv	32x32x3x200	5x5x3x384	1	2	32x32x384x200
2	relu	32x32x384x200	Max(0,x)	1	0	32x32x384x200
3	conv	32x32x384x200	1x1x384x320	1	0	32x32x320x200
4	relu	32x32x320x200	Max(0,x)	1	0	32x32x320x200
5	conv	32x32x320x200	1x1x320x192	1	0	32x32x192x200
6	relu	32x32x192x200	Max(0,x)	1	0	32x32x192x200
7	pool (max)	32x32x192x200	3x3	2	0	15x15x192x200
8	dropout (0.5)	15x15x192x200				15x15x192x200
9	conv	15x15x192x200	5x5x192x384	1	2	15x15x384x200
10	relu	15x15x384x200	Max(0,x)	1	0	15x15x384x200
11	conv	15x15x384x200	1x1x384x384	1	0	15x15x384x200
12	relu	15x15x384x200	Max(0,x)	1	0	15x15x384x200
13	conv	15x15x384x200	1x1x384x384	1	0	15x15x384x200
14	relu	15x15x384x200	Max(0,x)	1	0	15x15x384x200
15	conv	15x15x384x200	1x1x384x384	1	0	15x15x384x200
16	relu	15x15x384x200				15x15x384x200
17	pool (avg)	15x15x384x200	3x3	2	0	7x7x384x200
18	dropout (0.55)	7x7x384x200				7x7x384x200
19	conv	7x7x384x200	5x5x384x384	1	2	7x7x384x200
20	relu	7x7x384x200				7x7x384x200
21	conv	7x7x384x200	1x1x384x384	1	0	7x7x384x200
22	relu	7x7x384x200				7x7x384x200
23	conv	7x7x384x200	1x1x384x384	1	0	7x7x384x200
24	relu	7x7x384x200				7x7x384x200
25	pool (max)	7x7x384x200	3x3	2	0	3x3x384x200
26	dropout (0.55)	3x3x384x200				3x3x384x200
27	conv	3x3x384x200	3x3x384x384	1	1	3x3x384x200
28	relu	3x3x384x200				3x3x384x200
29	conv	3x3x384x200	1x1x384x384	1	0	3x3x384x200
30	relu	3x3x384x200				3x3x384x200
31	conv	3x3x384x200	1x1x384x200	1	0	3x3x200x200
32	pool (avg)	3x3x200x200	3x3	1	0	1x1x200x200
33	softmaxloss	1x1x200x200		1	0	1x1

Anish Shenoy (0.638)

- Batch normalization
- Xavier initialization
- Learning rate tuning

Table 8: Deep network 3

Layer	type	Input	Filter	Stride	Pad	Output
Layer 1	Conv	32x32x3	3x3x3x64	1	1	32x32x64
Layer 2	Relu	32x32x64	max(0,x)	1	0	32x32x64
Layer 3	BatchNorm	32x32x64	BatchNorm	1	0	32x32x64
Layer 4	Dropout	32x32x64	Drop p=0.5	1	0	32x32x64
Layer 5	Conv	32x32x64	3x3x64x64	1	1	32x32x64
Layer 6	Relu	32x32x64	max(0,x)	1	0	32x32x64
Layer 7	BatchNorm	32x32x64	BatchNorm	1	0	32x32x64
Layer 8	Max pool	32x32x64	2x2	2	0	16x16x64
Layer 9	Dropout	16x16x64	Drop p=0.5	1	0	16x16x64
Layer 10	Conv	16x16x64	3x3x64x128	1	1	16x16x128
Layer 11	Relu	16x16x128	max(0,x)	1	0	16x16x128
Layer 12	BatchNorm	16x16x128	BatchNorm	1	0	16x16x128
Layer 13	Dropout	16x16x128	Drop p=0.5	1	0	16x16x128
Layer 14	Conv	16x16x128	3x3x128x128	1	1	16x16x128
Layer 15	Relu	16x16x128	max(0,x)	1	0	16x16x128
Layer 16	BatchNorm	16x16x128	BatchNorm	1	0	16x16x128
Layer 17	Max pool	16x16x128	2x2	2	0	8x8x128
Layer 9	Dropout	8x8x128	Drop p=0.5	1	0	8x8x128
Layer 10	Conv	8x8x128	3x3x128x256	1	1	8x8x256
Layer 11	Relu	8x8x256	max(0,x)	1	0	8x8x256
Layer 12	BatchNorm	8x8x256	BatchNorm	1	0	8x8x256
Layer 13	Dropout	8x8x256	Drop p=0.5	1	0	8x8x256
Layer 14	Conv	8x8x256	3x3x256x256	1	1	8x8x256
Layer 15	Relu	8x8x256	max(0,x)	1	0	8x8x256
Layer 16	BatchNorm	8x8x256	BatchNorm	1	0	8x8x256
Layer 17	Max pool	8x8x256	2x2	2	0	4x4x256
Layer 18	Dropout	4x4x256	Drop p=0.5	1	0	4x4x256
Layer 19	Conv	4x4x256	4x4x256x500	1	0	1x1x500
Layer 20	Relu	1x1x500	max(0,x)	1	0	1x1x500
Layer 21	BatchNorm	1x1x500	BatchNorm	1	0	1x1x500
Layer 22	Dropout	1x1x500	Drop p=0.5	1	0	1x1x500
Layer 23	Conv	1x1x500	1x1x500x100	1	0	1x1x100
Layer 24	Softmax loss	1x1x100		1	0	1x100

Today’s class

- Review of important concepts
- Some important open problems
- Feedback and course evaluation

Fundamentals of Computer Vision

- Light
  - What an image records
- Geometry
  - How to relate world coordinates and image coordinates
- Matching
  - How to measure the similarity of two regions
- Alignment
  - How to align points/patches
  - How to recover transformation parameters based on matched points
- Grouping
  - What points/regions/lines belong together?
- Categorization
  - What similarities are important?

Light and Color

- Shading of diffuse materials depends on albedo and orientation wrt light
  - Gradients are a major cue for changes in orientation (shape)
- Many materials have a specular component that directly reflects light
- Reflected color depends on albedo and light color
- RGB is default color space, but sometimes others (e.g., HSV, L*a*b) are more useful

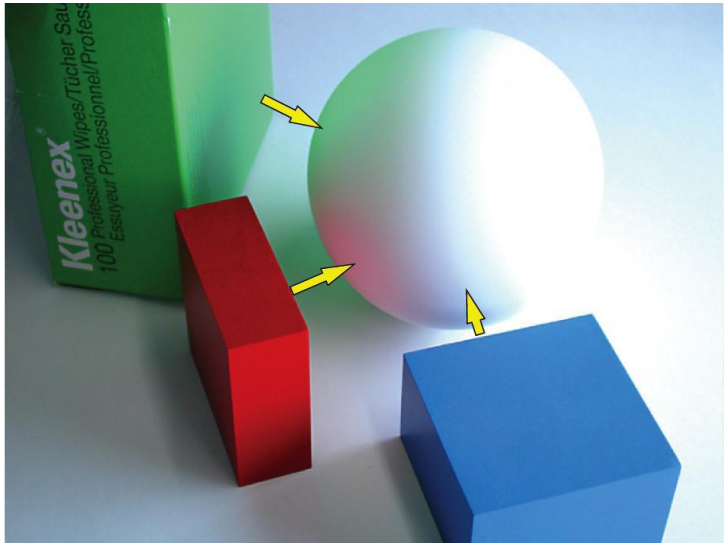


Image from Koenderink

Geometry

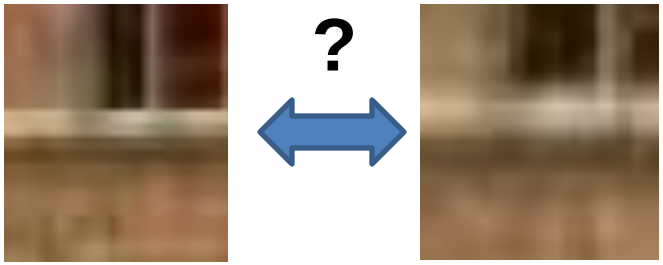
- $\mathbf{x} = \mathbf{K}\,[\mathbf{R}\ \mathbf{t}]\,\mathbf{X}$
  - Maps 3d point $\mathbf{X}$ to 2d point $\mathbf{x}$
  - Rotation $\mathbf{R}$ and translation $\mathbf{t}$ map into camera's 3D coordinates
  - Intrinsic matrix $\mathbf{K}$ projects from 3D to 2D

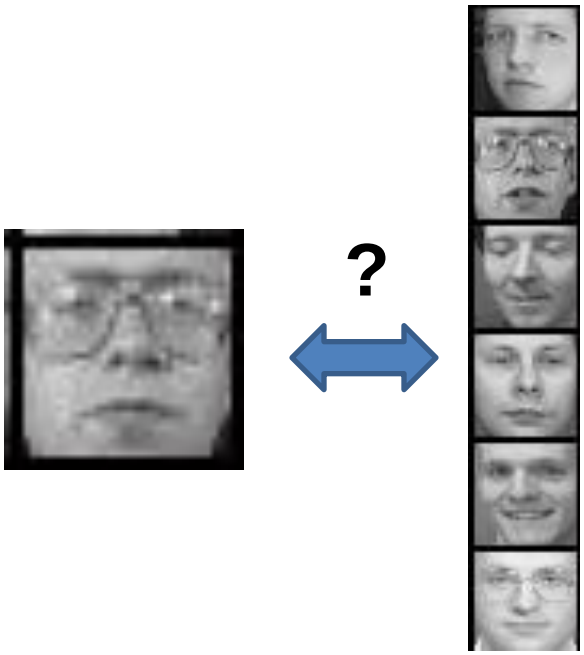
- Parallel lines in 3D converge at the **vanishing point** in the image
  - A 3D plane has a vanishing line in the image

- $\mathbf{x'}^{\mathbf{T}}\,\mathbf{F}\,\mathbf{x} = 0$
  - Points in two views that correspond to the same 3D point are related by the fundamental matrix $\mathbf{F}$

Matching

- Does this patch match that patch?
  - In two simultaneous views? (stereo)
  - In two successive frames? (tracking, flow, SFM)
  - In two pictures of the same object? (recognition)





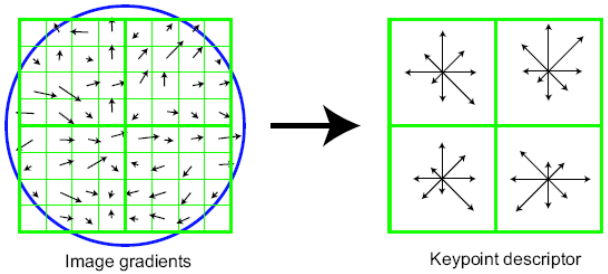
Matching

Representation: be invariant/robust to expected deformations but nothing else

- Assume that shape does not change
  - Key cue: local differences in shading (e.g., gradients)
- Change in viewpoint
  - Rotation invariance: rotate and/or affine warp patch according to dominant orientations
- Change in lighting or camera gain
  - Average intensity invariance: oriented gradient-based matching
  - Contrast invariance: normalize gradients by magnitude
- Small translations
  - Translation robustness: histograms over small regions

But can one representation do all of this?

- SIFT: local normalized histograms of oriented gradients provides robustness to in-plane orientation, lighting, contrast, translation
- HOG: like SIFT but does not rotate to dominant orientation



Alignment of points

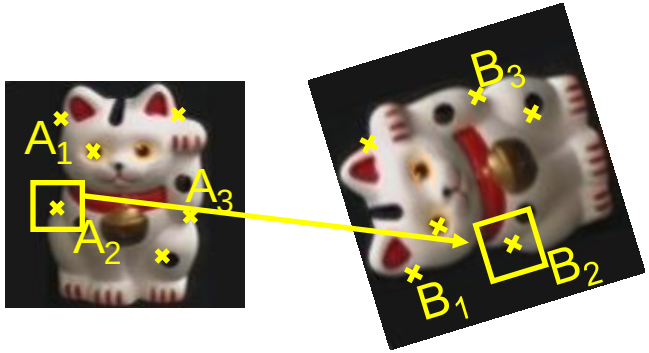
Search: efficiently align matching patches

- Interest points: find repeatable, distinctive points
  - Long-range matching: e.g., wide baseline stereo, panoramas, object instance recognition
  - Harris: points with strong gradients in orthogonal directions (e.g., corners) are precisely repeatable in x-y
  - Difference of Gaussian: points with peak response in Laplacian image pyramid are somewhat repeatable in x-y-scale
- Local search
  - Short range matching: e.g., tracking, optical flow
  - Gradient descent on patch SSD, often with image pyramid
- Window/scan search
  - Long-range matching: e.g., recognition, stereo w/ scanline

Alignment of sets

Find transformation to align matching sets of points

- Geometric transformation (e.g., affine)
  - Least squares fit (SVD), if all matches can be trusted
  - Hough transform: each potential match votes for a range of parameters
    - Works well if there are very few parameters (3-4)
  - RANSAC: repeatedly sample potential matches, compute parameters, and check for inliers
    - Works well if fraction of inliers is high and few parameters (4-8)
- Other cases
  - Thin plate spline for more general distortions
  - One-to-one correspondence (Hungarian algorithm)



Grouping

- Clustering: group items (patches, pixels, lines, etc.) that have similar appearance
  - Uses: discretize continuous values; improve efficiency; summarize data
  - Algorithms: k-means, agglomerative

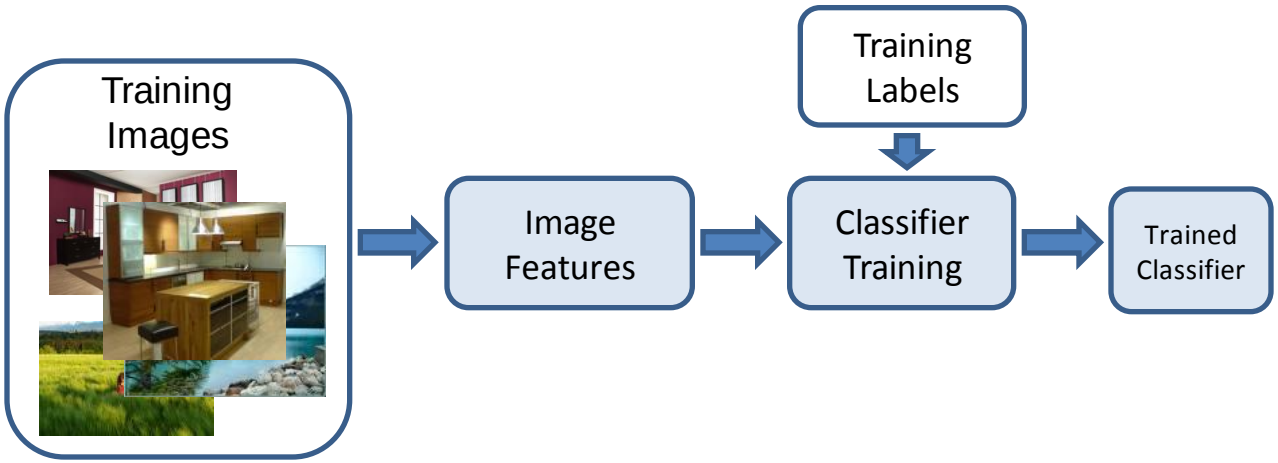
- Segmentation: group pixels into regions of coherent color, texture, motion, and/or label
  - Mean-shift clustering
  - Watershed
  - Graph-based segmentation: e.g., MRF and graph cuts

- EM, mixture models: probabilistically group items that are likely to be drawn from the same distribution, while estimating the distributions' parameters

Recognition

Match objects, parts, or scenes that may vary in appearance

- Categories are typically defined by human and may be related by function, cost, or other non-visual attributes
- Key problem: what are important similarities?
  - Can be learned from training examples



Recognition

- Major improvements in feature learning (CNNs) over past five years
  - If you have lots (100K+) examples, you can learn features from scratch
  - Otherwise, use outputs from a pre-trained network
- Similar networks can be used for object detection, pose estimation, and segmentation

Vision as part of an intelligent system

3D Scene

Feature Extraction

- Texture
- Color
- Optical Flow
- Stereo Disparity

Grouping

- Surfaces
- Bits of objects
- Sense of depth
- Motion patterns

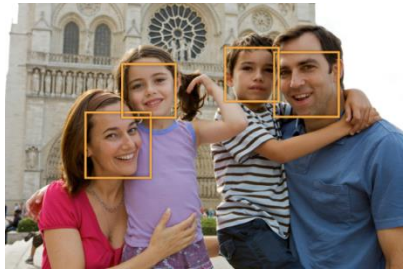
Interpretation

- Objects
- Agents and goals
- Shapes and properties
- Open paths
- Words

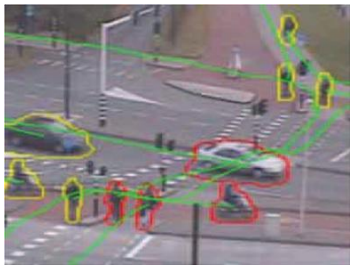
Action

- Walk, touch, contemplate, smile, evade, read on, pick up, …

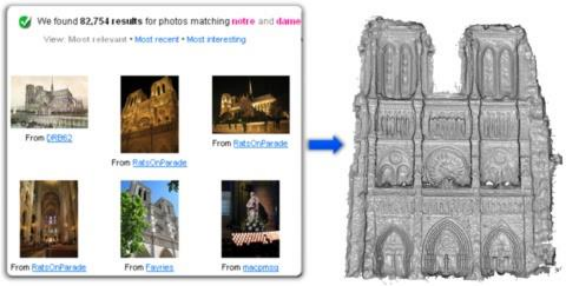
Well-Established (patch matching)



Face Detection/Recognition



Object Tracking / Flow

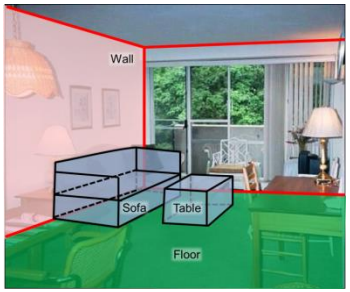


Multi-view Geometry

Major Progress (pattern matching++)

Category Detection

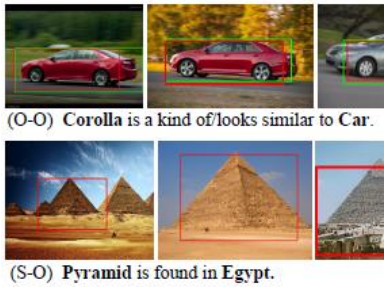
Human Pose



3D Scene Layout

New Opportunities (interpretation/tasks)

Entailment/Prediction



Life-long Learning

Vision for Robots

Vision is ready to break out of the lab

Vision is not the goal, but a way to learn about your immediate environment and the world, so that you can act.

Scene Understanding =

Objects + People + Layout +

Interpretation *within Task Context*

What do I see? → Why is this happening?

What is important?

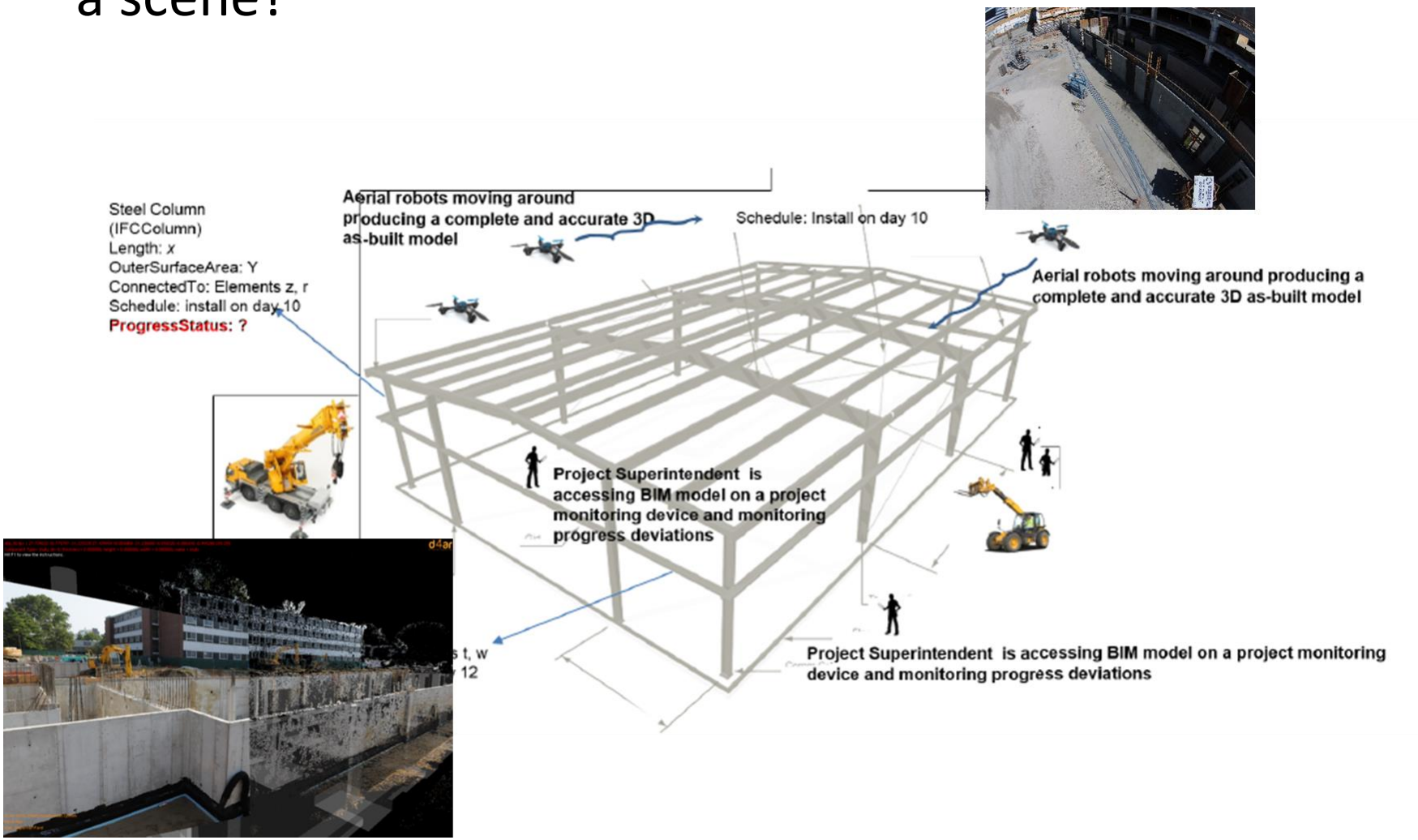
What will I see?

How can we learn about the world through vision?

How do we create/evaluate vision systems that adapt to useful tasks?

Important open problems

- How can we interpret vision given structured plans of a scene?



Important open problems

- Algorithms: works pretty well → nearly perfect
  - Stereo: top of wish list from Pixar guy Micheal Kass
  - Structure-from-motion

Good directions:

- Incorporate higher level knowledge

Important open problems

- Spatial understanding

Important questions:

- What are good representations of space for navigation and interaction? What kind of details are important?
- How can we combine single-image cues with multi-view cues?

Important open problems

Object representation: what is it?

Important questions:

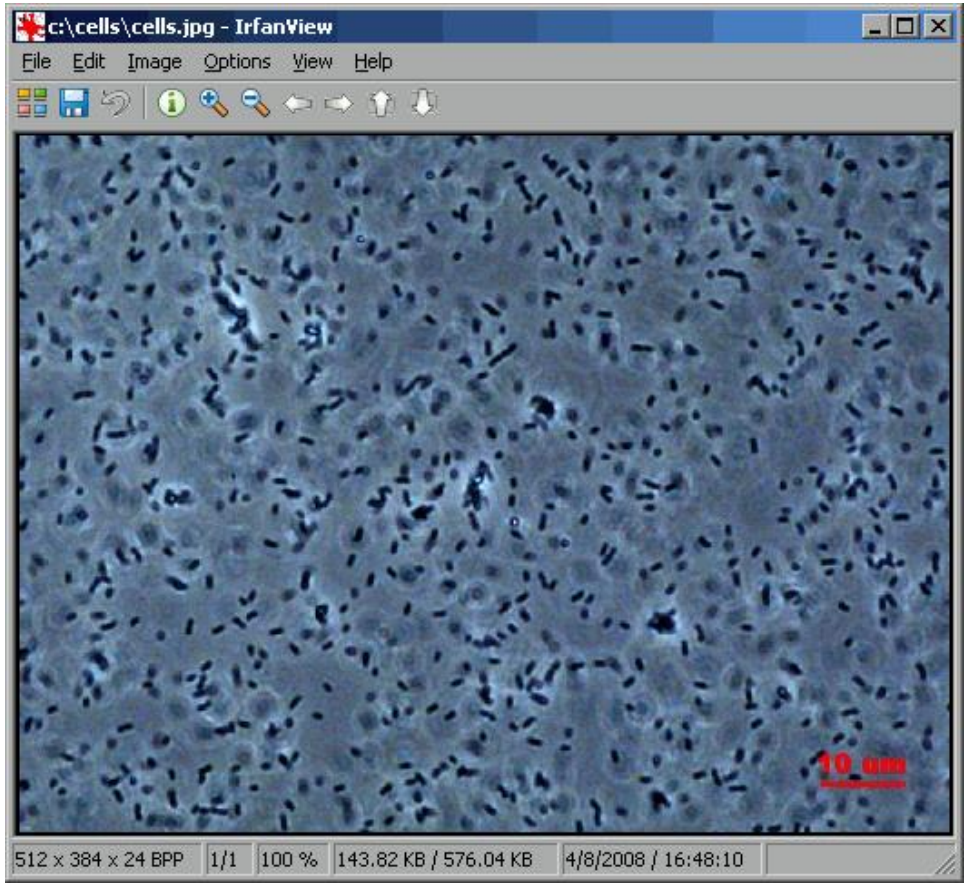
- How can we pose recognition so that it lets us deal with new objects?
- What do we want to predict or infer, and to what extent does that rely on categorization?
- How do we transfer knowledge of one type of object to another?

Important open problems

- Can we build a “core” vision system that can easily be extended to perform new tasks or even learn on its own?
  - What kind of representations might allow this?
  - What should be built in and what should be learned?

Important open problems

- Vision for the masses



Counting cells

Analyzing social effects of green space

How to make vision systems that can quickly adapt to these thousands of visual tasks?

If you want to learn more…

- Read lots of papers: IJCV, PAMI, CVPR, ICCV, ECCV, NIPS
- Helpful topics for classes
  - David Forsyth's optimization
  - Classes in machine learning or pattern recognition
  - Statistics, graphical models
  - Seminar-style paper-reading classes
- Just implement stuff, try demos, see what works

Feedback through Google Form

More specific feedback to help improve course

https://goo.gl/forms/FIfUqnVH8pcGHoCe2

ICES Forms

- Looking forward to your project presentations!