Computer Vision: Summary and Discussion

Computer Vision

CS 543 / ECE 549

University of Illinois

Derek Hoiem

HW 5

- 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

- What image categorization approaches worked best?
  - Wan Chen: Gist + SVM = 87.8% accuracy

Dataset creators (Oliva and Torralba) report 84% with Gist + RBF-SVM (different train/test split)

HW 5

- What image categorization approaches worked best?
  - Huy Le

Models	HSV histogram pyramid	BOW pyramid	GIST Descriptor	CENTRIST descriptor
Vector Dimension	7560	500	512	254
Training time(s)	126.502042	3.511960	3.276901	1.653669
Testing time(s)	46.561153	0.505256	0.400885	0.280098
Accuracy	61.12%	68.75%	87.38%	82%

Combined: 91.4%

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

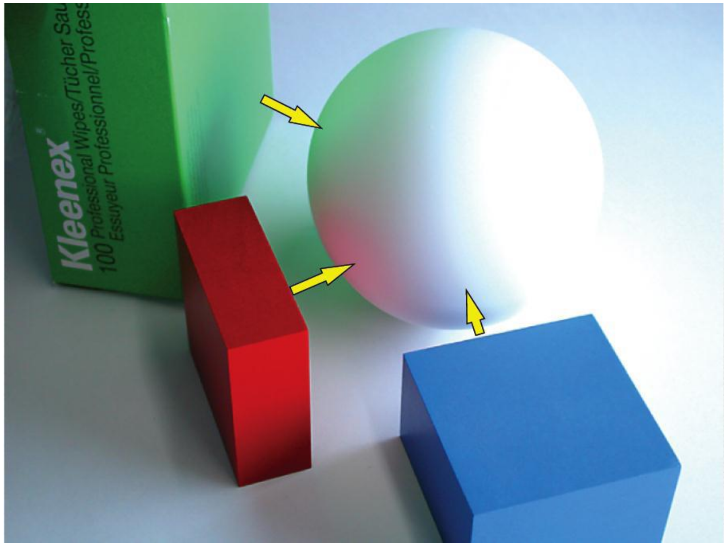


Image from Koenderink

Geometry

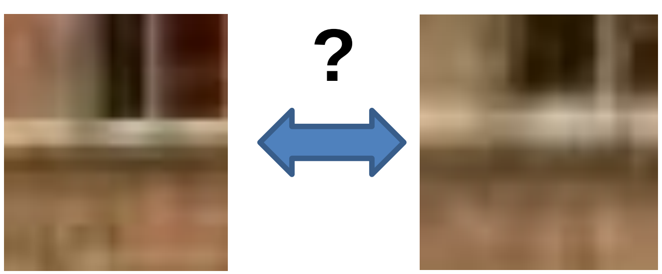
- $\mathbf{x} = \mathbf{K}\,[\mathbf{R}\ \mathbf{t}]\,\mathbf{X}$
  - Maps 3d point $\mathbf{X}$ to 2d point $\mathbf{x}$
  - Rotation $\mathbf{R}$ and translation $\mathbf{t}$ map into 3D camera coordinates
  - Intrinsic matrix $\mathbf{K}$ projects from 3D to 2D

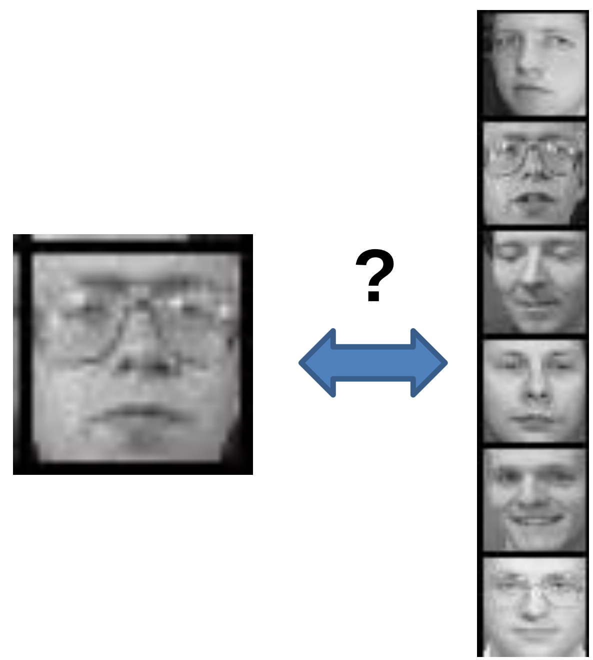
- Parallel lines in 3D converge at the **vanishing point** in the image
  - A 3D plane has a vanishing line in the image

- $\mathbf{x}'^{\mathrm{T}}\,\mathbf{F}\,\mathbf{x} = 0$
  - 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)



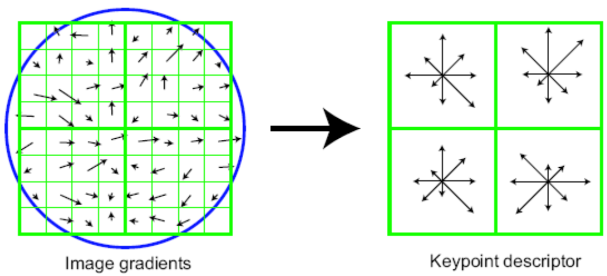


Matching

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

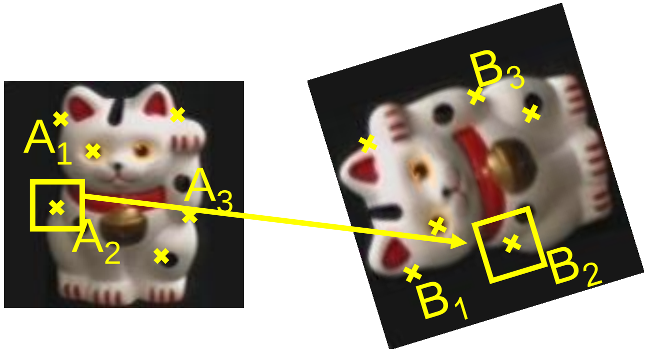
Search: efficiently align matching patches

- 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
- Windowed 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

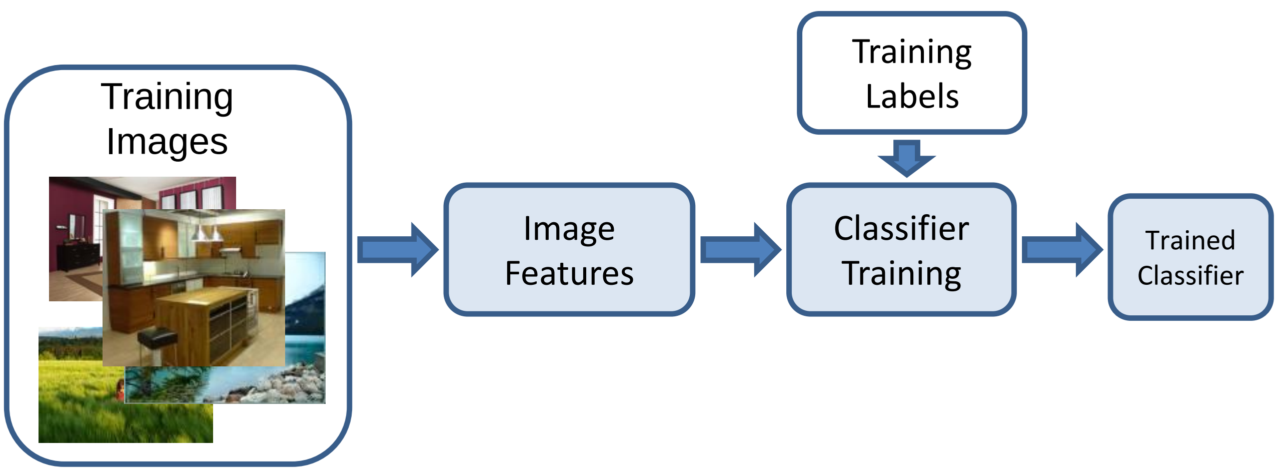
- Segmentation: group pixels into regions of coherent color, texture, motion, and/or label
  - Mean-shift clustering
  - Watershed
  - Graph-based segmentation: e.g., MRF and graph cuts

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

Categorization

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



Categorization

Representation: ideally should be compact, comprehensive, direct

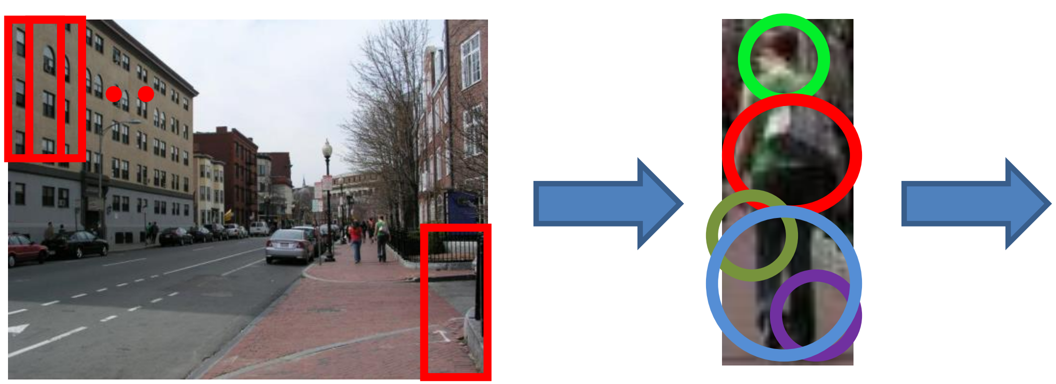
- Histograms of quantized interest points (SIFT, HOG), color, texture
  - Typical for image or region categorization
  - Degree of spatial encoding is controllable by using spatial pyramids
- HOG features at specified position
  - Often used for finding parts or objects

Object Categorization

Search by Sliding Window Detector

- May work well for rigid objects

- Key idea: simple alignment for simple deformations



Object or Background?

Object Categorization

Search by Parts-based model

- Key idea: more flexible alignment for articulated objects
- Defined by models of **part appearance**, **geometry** or spatial layout, and **search** algorithm

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

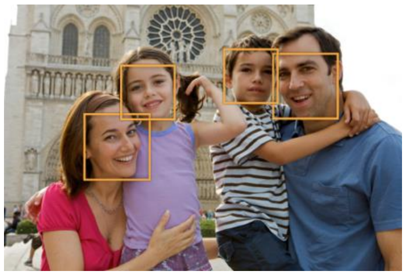
Interpretation

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

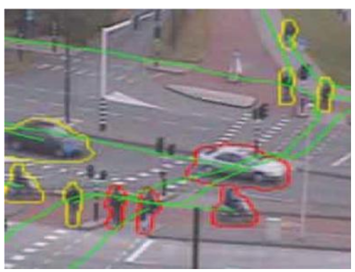
Action

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

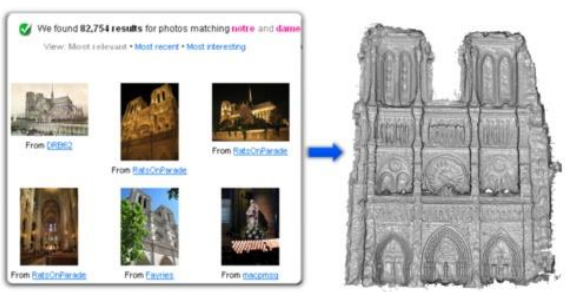
Well-Established (patch matching)



Face Detection/Recognition



Object Tracking / Flow

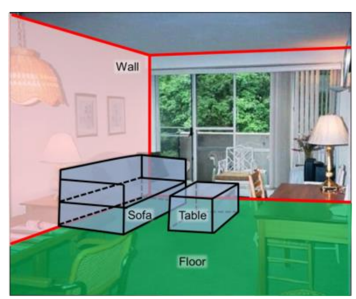


Multi-view Geometry

Major Progress (pattern matching++)

Category Detection

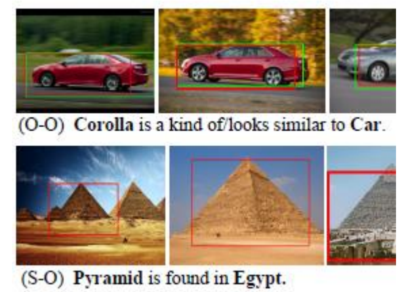
Human Pose



3D Scene Layout

New Opportunities (interpretation/tasks)

Entailment/Prediction



Life-long Learning

Vision for Robots

Scene Understanding =
Objects + People + Layout +
Interpretation *within Task Context*

What do I see? → Why is this happening?

What is important?

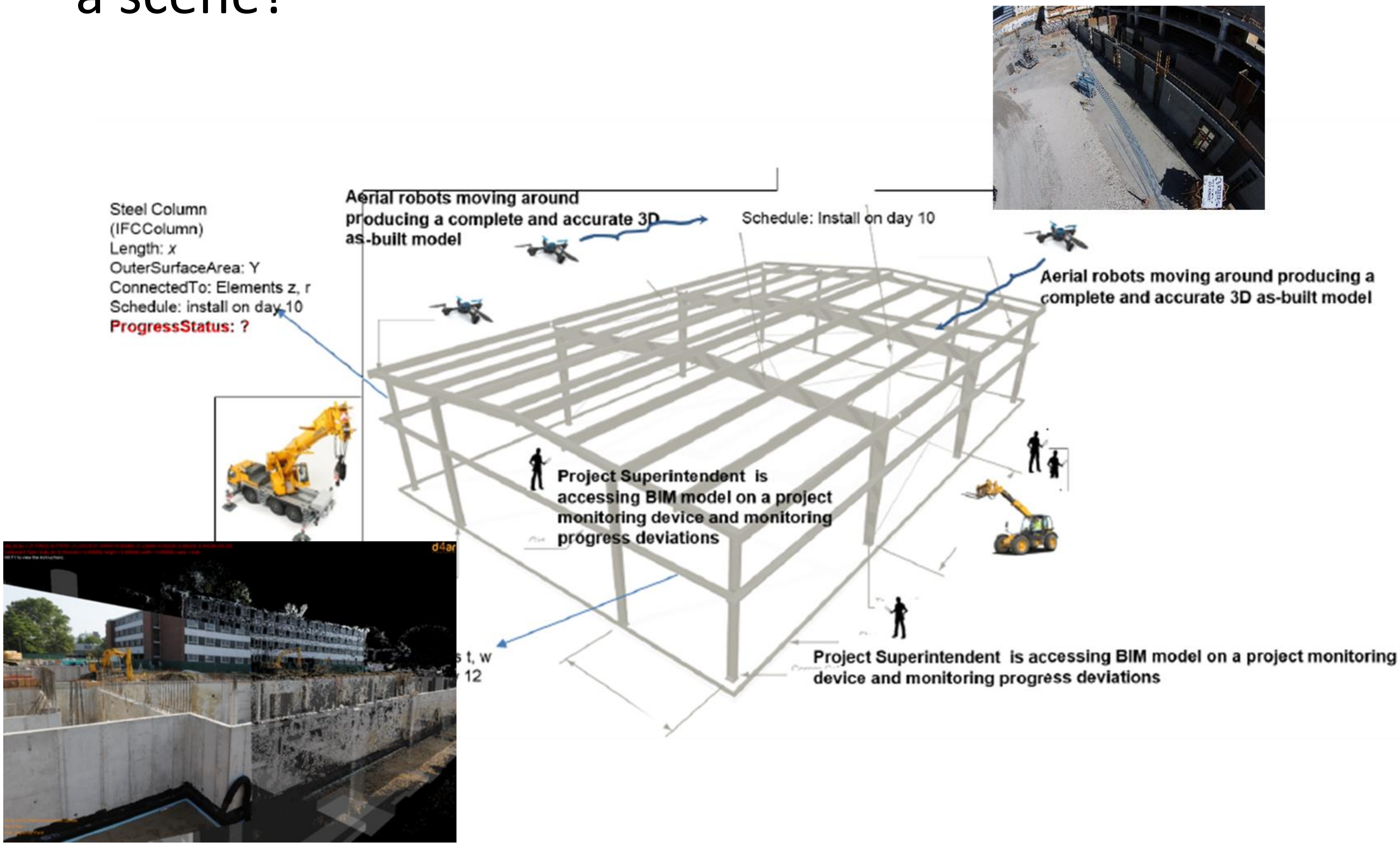
What will I see?

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 → perfect
  - E.g., stereo: top of wish list from Pixar guy Micheal Kass

Good directions:

- Incorporate higher level knowledge

Important open problems

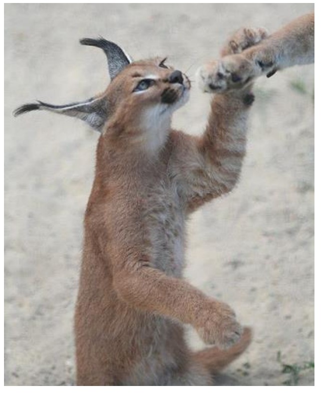
- Spatial understanding: what is it doing? Or how do I do it?

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:

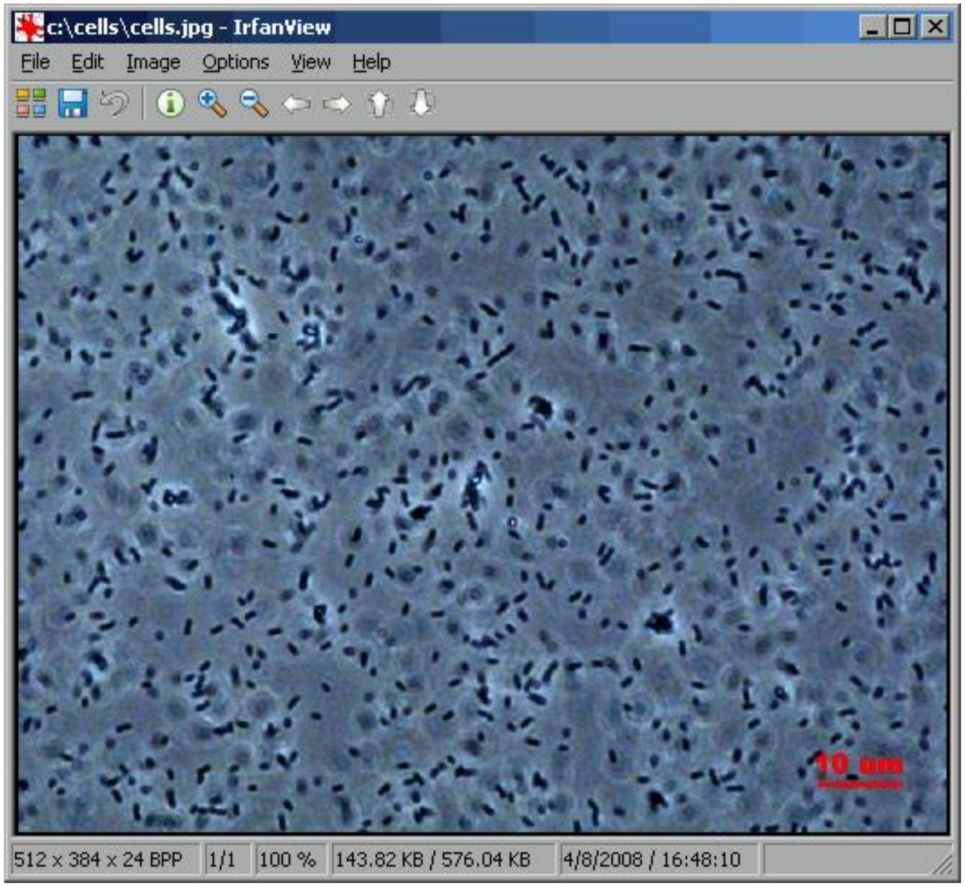
- How can we pose recognition so that it lets us deal with new objects?
- What do we want to predict or infer, and to what extent does that rely on categorization?
- How do we transfer knowledge of one type of object to another?

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?

Important problems

- Learning features and intermediate representations that transfer to new tasks

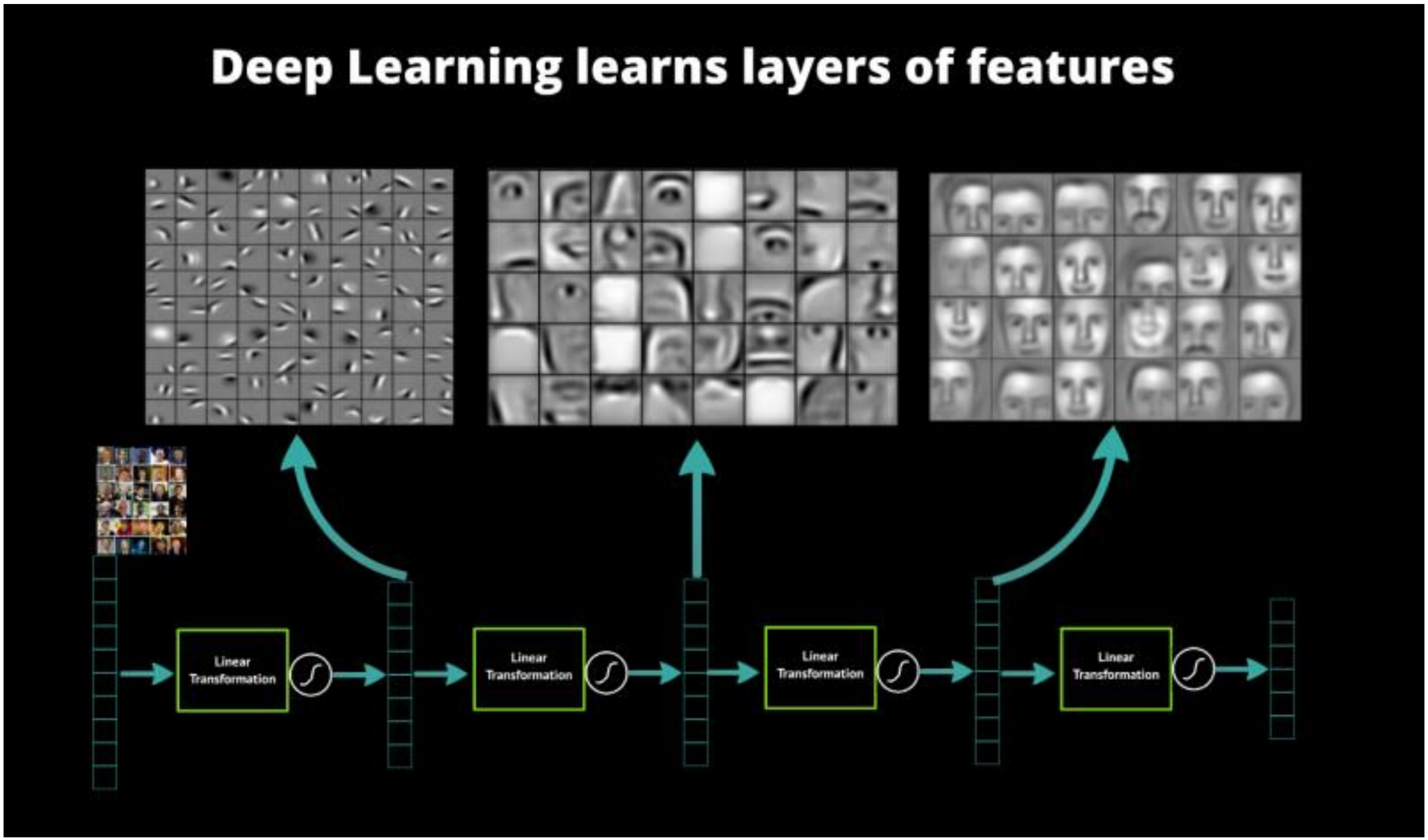


Figure from http://www.datarobot.com/blog/a-primer-on-deep-learning/

Important open problems

- Almost everything is still unsolved!
  - Robust 3D shape from multiple images
  - Recognize objects (only faces and maybe cars is really solved, thanks to tons of data)
  - Caption images/video
  - Predict intention
  - Object segmentation
  - Count objects in an image
  - Estimate pose
  - Recognize actions
  - ….

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

ICES Forms: very important

- See you next week!