Advanced Image Processing
(aka *Inverse Problems in Image Processing*)
CS 754

Ajit Rajwade,
IITB, CSE Department
Spring 2025

Why take this course?

- This course covers some advanced topics in image processing and image acquisition.
- The purpose of this course is to introduce you to some of the **frontiers** of the image processing field.
- It will cover mostly very **contemporary** topics (that have been published in the last 10-15 years).
- Will be useful to **machine learning, statistics** or **signal processing** people as well.
- Useful for people in CSE, EE, CSRE, Earth Sciences, EP, Mathematics, CMiNDS, KCDH, IEOR, etc.

Why take this course?

- Image Processing is an inherently **interdisciplinary** subject: numerous application areas - remote sensing, photography, visual psychology, archaeology, surveillance, etc.

- Has become a very popular field of study in India: scope for R&D work in numerous research labs (In India: GE, Phillips, Siemens, Microsoft, HP, TI, Google; DRDO, ICRISAT, ISRO, etc.)

Why take this course?

- India has numerous conferences in image processing and related areas: ICVGIP, NCVPRIPG, SPCOM, NCC.
- International conferences in this area: CVPR, ICCV, ECCV, ICIP, ICASSP, MMSP and many more.
- Image Processing papers are to be found in many machine learning conferences as well – eg NIPS, ICML.

Why take this course?

- One of the recommended courses if you want to do research in image processing.

- You will get to work on a nice **course project**!

Computer Vision and Image Processing: What's the difference?

- Difference is **blurry**
- "Image processing" typically involves processing/analysis of (2D) images without referring to underlying 3D structure
- Computer vision – typically involves inference of **underlying 3D structure** from 2D images
- Many computer vision techniques also aim to infer properties of the scene directly – without 3D reconstruction.
- Computer vision – direct opposite of computer graphics

This course is…

- It’s not a computer vision course
- It’s not a graphics or animation course
- It’s not a medical imaging course
- It’s not a course on mathematics
- This course covers a different set of topics as compared to CS 663 (the basic image processing course)
- You can (and are recommended to) take CS 663 even after taking CS 754.

Course web-page

http://www.cse.iitb.ac.in/~ajitvr/CS754_Spring2025/

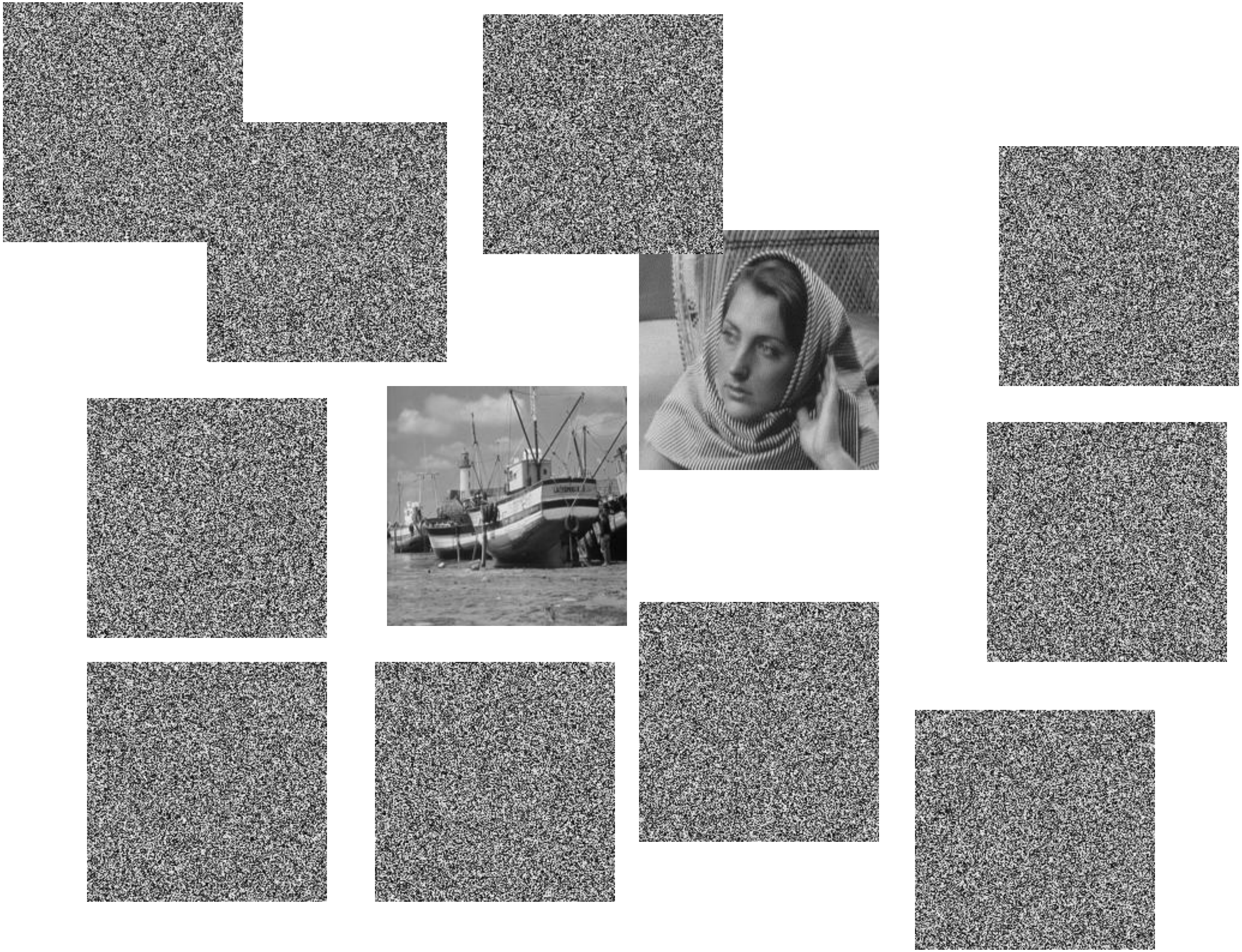
What will we study in this course?

Major components of course syllabus

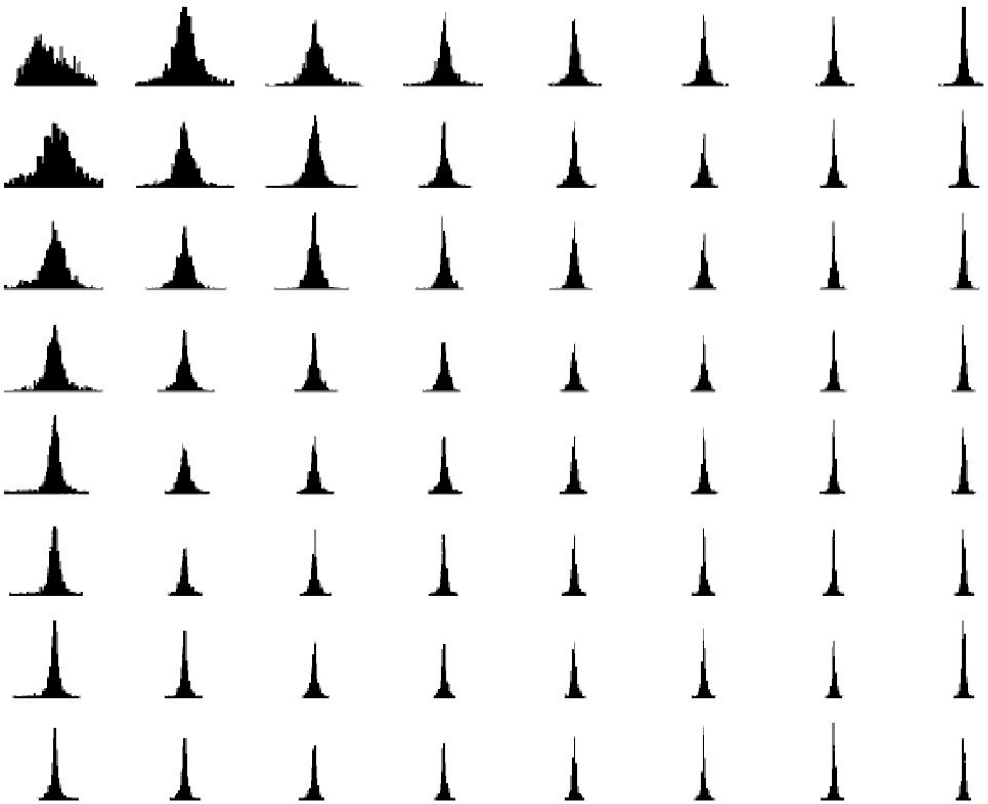
- Compressed Sensing
- Tomography
- Learning image representations: dictionary learning, transform learning
- Low rank matrix recovery and completion, robust principal components analysis
- Brief coverage: Statistics of natural images and textures
- Concept of phase retrieval (time permitting)
- **Applications in inverse problems:** image denoising, image deblurring, image category classification, reflection removal, forensics, and many others.

Statistics of Natural Images

- Number of possible 200 x 200 images (of 256, i.e. 8 bit intensity levels) = 256^40000 = 2^320000 = 10^110000.
- This is several trillion times the number of atoms in the universe (10^90).
- Only a tiny subset of these are plausible as natural images.



Statistics of Natural Images: example



Histograms of DCT coefficients
of small image patches

https://pdfs.semanticscholar.org/e41f/fe0a1462eb0117e56c1570e3cf7e1bc8b5eb.pdf

Statistics of Natural Images: example

Image source: Buccigrossi et al, Image Compression via Joint Statistical Characterization in the Wavelet Domain

Fig. 3. Coefficient magnitudes of a wavelet decomposition. Shown are absolute values of subband coefficients at three scales, and three orientations of a separable wavelet decomposition of the Einstein image. Also shown is the lowpass residual subband (upper left). Note that high-magnitude coefficients of the subbands tend to be located in the same (relative) spatial positions.

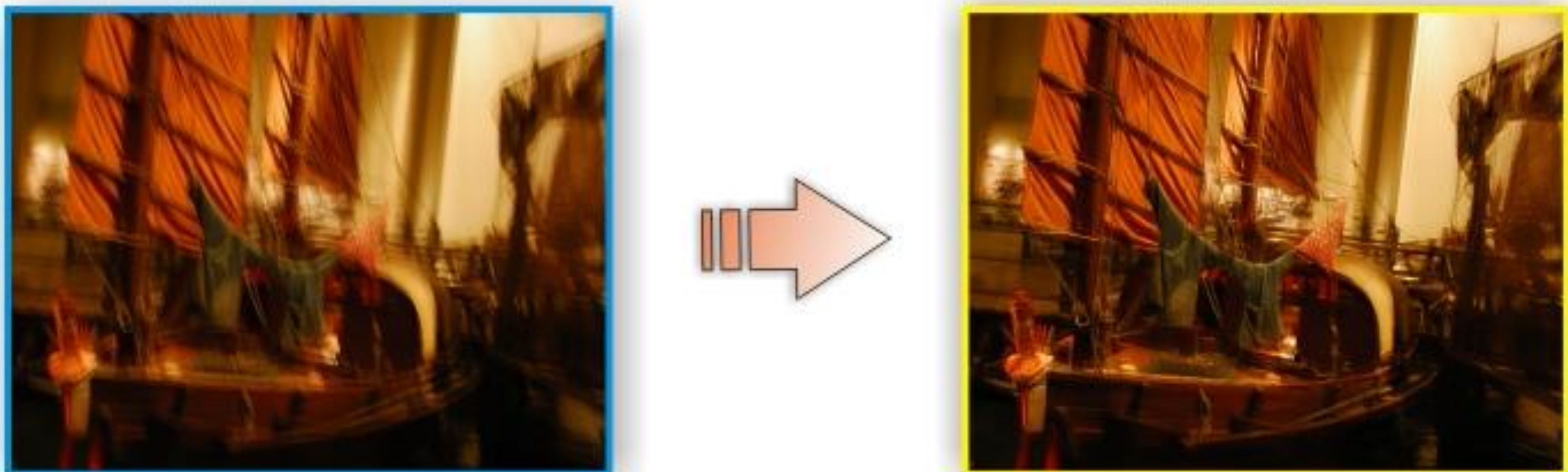
Large magnitude coefficients tend to occur at neighboring spatial locations within a sub-band, or at the same locations in sub-bands of adjacent scale/orientation

Applications of these properties

- Image denoising
- Image deblurring
- Image inpainting
- Image compression
- Image-based forensics
- Reflection removal

Sample State of the art result: Gaussian Noise sigma = 15

Motion deblurring



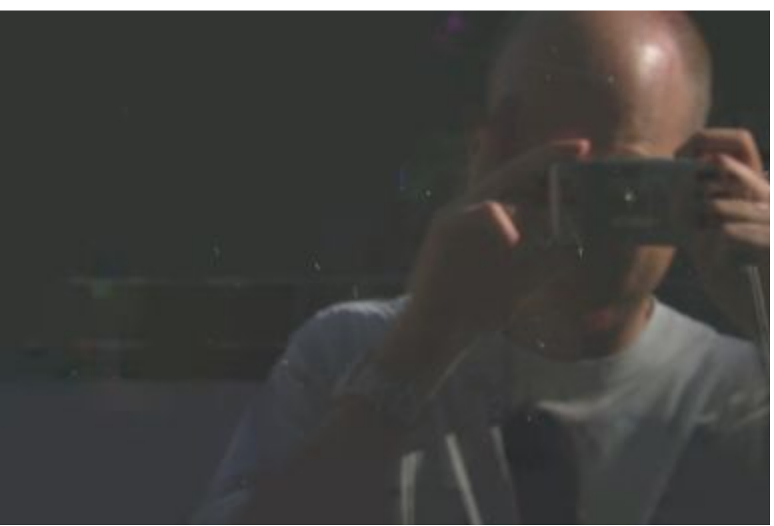
http://www.cse.cuhk.edu.hk/leojia/projects/motion_deblurring/

Inpainting

http://www.dtic.upf.edu/~mbertalmio/restorationo.html

Reflection Removal

http://webee.technion.ac.il/people/anat.levin/papers/Assisted-Reflections-Levin-Weiss-PAMI.pdf



Classification Problems

http://web.mit.edu/torralba/www/ne3302.pdf

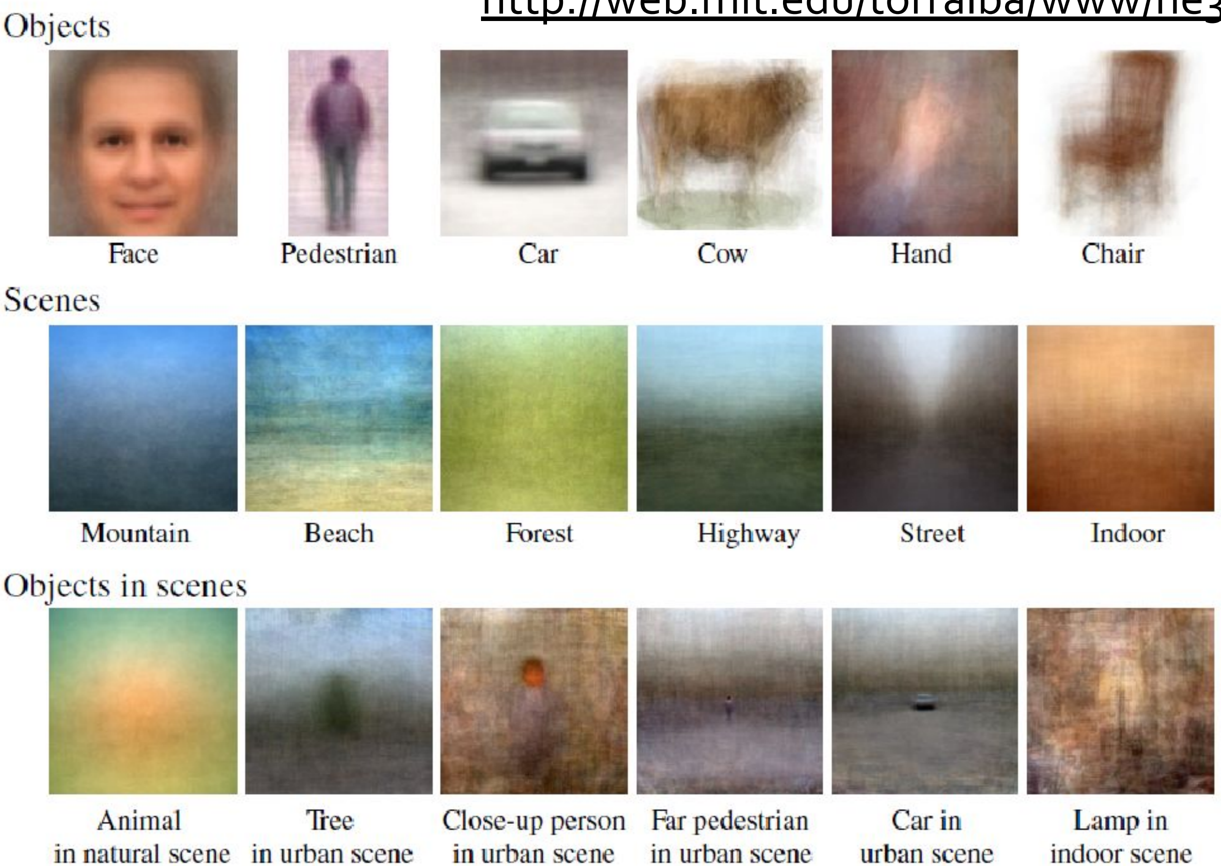
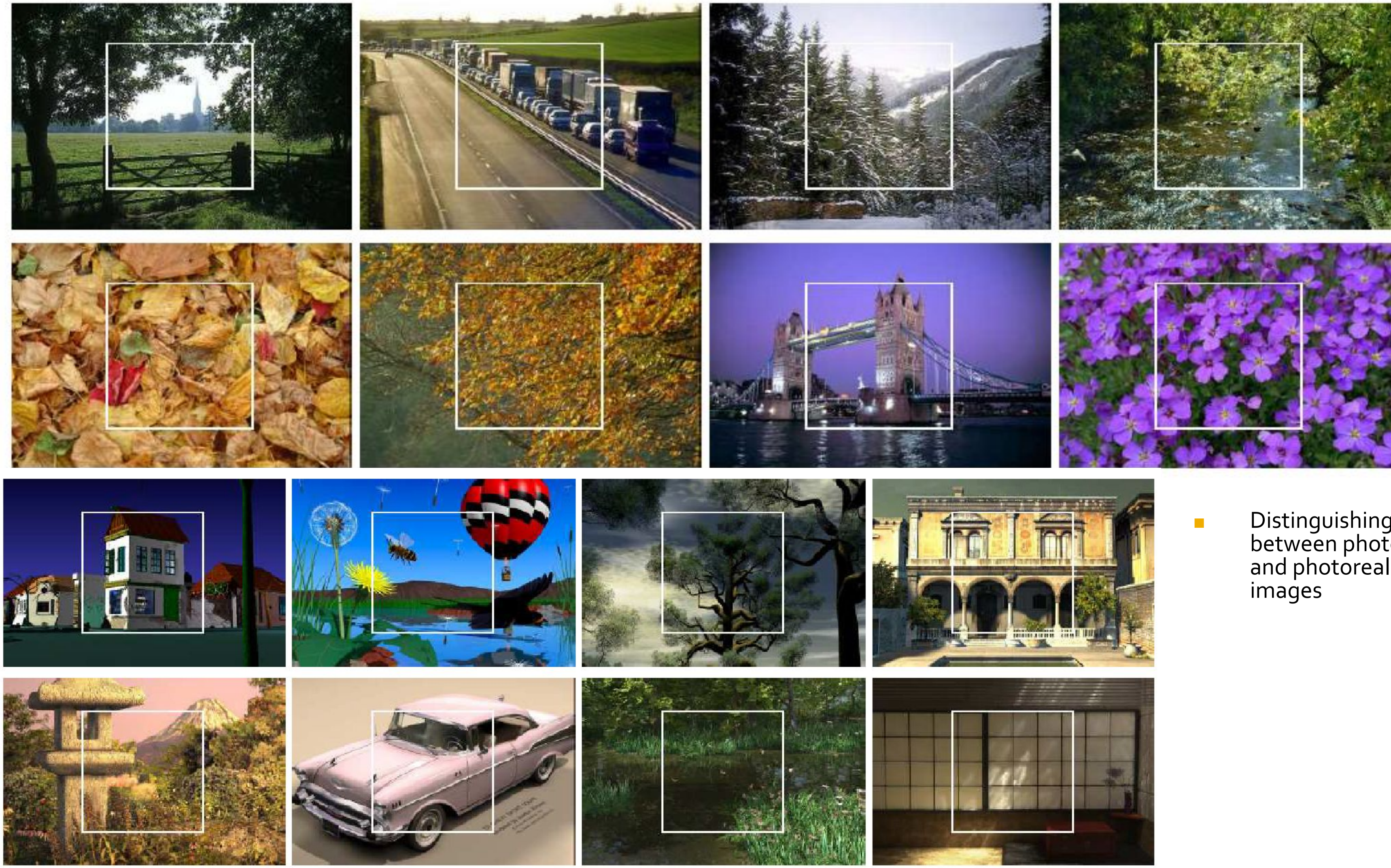


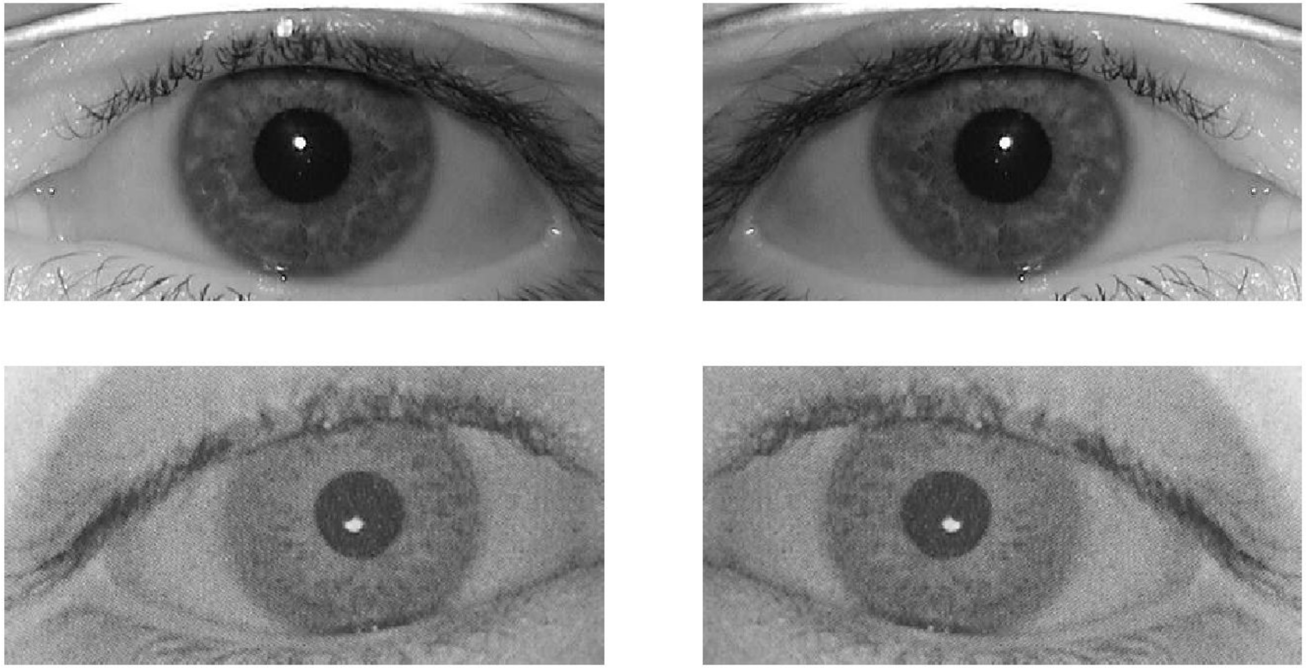
Figure 1. Averaged pictures of categories of objects, scenes and objects in scenes, computed with 100 exemplars or more per category. Exemplars were chosen to have the same basic level and viewpoint in regard to an observer. The group objects in scenes (third row) represent examples of the averaged peripheral information around an object centred in the image.

Classification Problems: Forensics



- Distinguishing between photographic and photorealistic images

Classification Problems: Forensics



- Distinguishing between live and rebroadcast images

Dictionary learning

- I have earlier told you that the DCT coefficients of image patches are sparse.
- This fact is aggressively used by the JPEG algorithm!
- So consider: $$y_i = \mathbf{U}\theta_i, 1 \leq i \leq N, \mathbf{U} = \text{2D DCT basis}, \theta_i \text{ is sparse}$$ $$y_i \in R^n, \theta_i \in R^n, \mathbf{U} \in R^{n \times n}$$
- Can you infer this **U** from the data instead of using the DCT basis?

Dictionary learning

- Can you infer this **U** from the data instead of using the DCT basis?
- You studied one such algorithm in CS 663 – it was PCA.
- It generated an orthonormal **U** matrix.
- It turns out that there are algorithms which do not require **U** to be orthonormal!
- And **U** not being orthonormal brings us many benefits!
- What are those benefits? We will study in detail in applications like compression and denoising.

Compressive Sensing

- In conventional sensing of images, the measurement device (i.e. camera) acquires a raw bitmap, and then compresses it using algorithms like JPEG (or MPEG in case of video).
- In compressive sensing, the measurement device acquires the image in a compressed format **directly by means of hardware**.
- Conversion from the compressed format to the conventional form is a challenging problem!

Compressive Sensing

$$\mathbf{y} = \mathbf{\Phi x}, \mathbf{y} \in \mathbf{R}^{\mathbf{m}}, \mathbf{x} \in \mathbf{R}^{\mathbf{n}}, \mathbf{\Phi} \in \mathbf{R}^{\mathbf{m} \times \mathbf{n}}, m << n$$

Compressive Measurement

Measurement matrix

Original image (in vectorised format)

Aim: to recover **x** given both **y** and **Φ.**

As *m* is much less than *n*, this problem is ill-posed in ordinary cases.

However if **x** and **Φ** obey certain properties (namely "Sparsity" and "incoherence" respectively), this problem becomes well-posed!

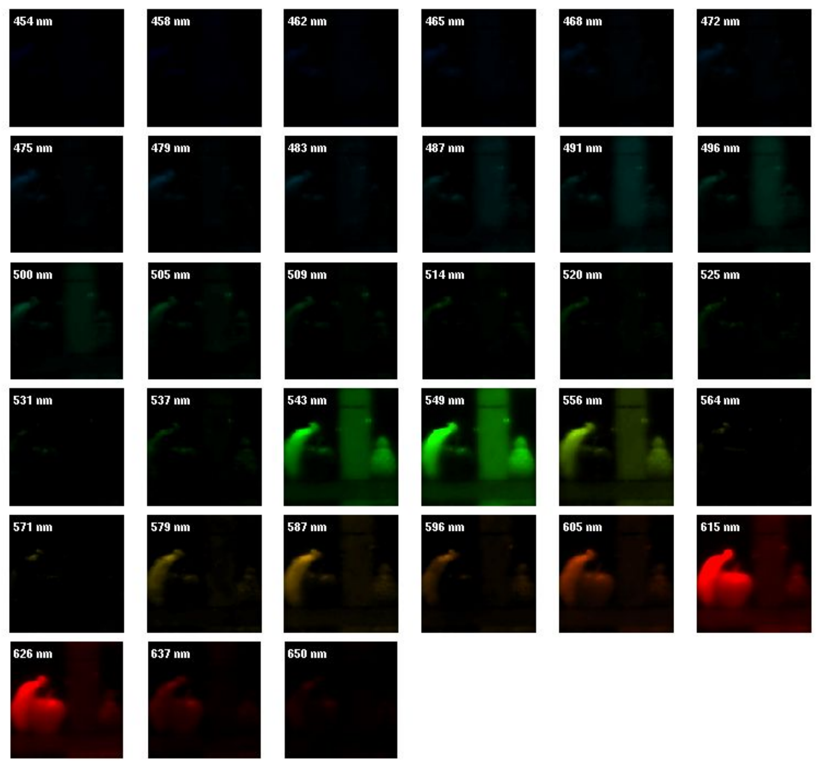
In fact, compressive sensing theory states that the recovery of **x** is almost perfect if these conditions are satisfied.

Compressive Sensing

- In this course, we will state the key theorems of compressive sensing.
- We will prove some of these theorems!
- We will look at algorithms for recovery of $\mathbf{x}$ given $\mathbf{y}$ and $\mathbf{\Phi}$.
- We will look at the architecture (block-diagram) of some existing compressive cameras – such as the Rice single pixel camera.

Compressive Sensing

- Applications will be explored in the areas of video acquisition, MRI and hyperspectral imagery.



http://www.disp.duke.edu/projects/CASSI/experimentaldata/index.ptml

Compressed Sensing: Success story!

- In MRI: https://t.co/30776nzj4T

Thanks to "compressed sensing" technology, which was developed in part at Rice, **scans of the beating heart can be completed in as few as 25 seconds** while the patient breathes freely. In contrast, in an MRI scanner equipped with conventional acceleration techniques, patients must lie still for four minutes or more and **hold their breath for as many as seven to 12 times** throughout a cardiovascular-related procedure.

Compressed Sensing: Success story!

- In video microscopy:
  https://link.springer.com/content/pdf/10.1186%2Fs40679-015-0009-3.pdf

  One of the main limitations of imaging at high spatial and temporal resolution during in-situ transmission electron microscopy (TEM) experiments is the frame rate of the camera being used to image the dynamic process. While the recent development of direct detectors has provided the hardware to achieve frame rates approaching 0.1 ms, the cameras are expensive and must replace existing detectors. In this paper, we examine the use of coded aperture compressive sensing (CS) methods to increase the frame rate of any camera with simple, low-cost hardware modifications. Depending on the resolution and signal/noise of the image, it should be possible to **increase the speed of any camera by more than an order of magnitude** using this approach.

Compressed Sensing for COVID-19 pooling

- COVID-19 has infected more than 85 crore people worldwide – India 1+ crore

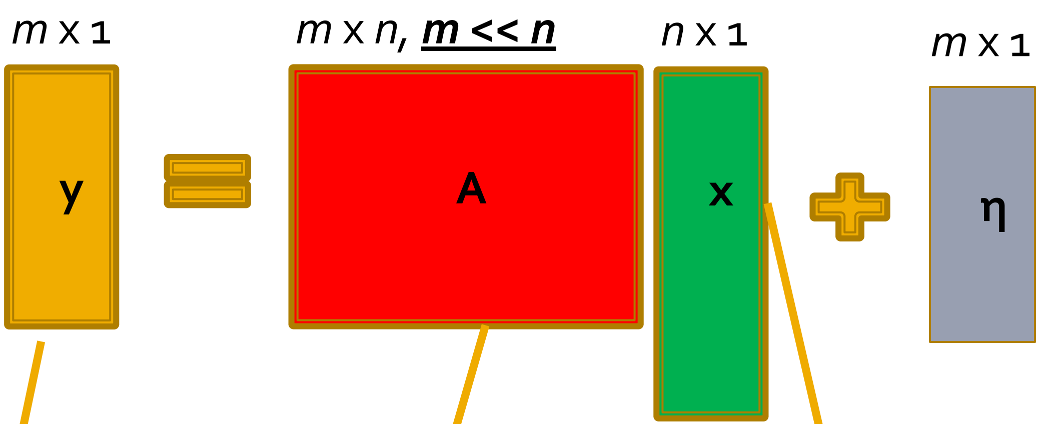
- RT-PCR is the most popular method for testing a person for COVID-19

- Dearth of resources for widespread testing: time, skilled manpower, reagents, testing kits, etc.

Compressed Sensing for COVID-19 pooling

- RT-PCR: naso- or oro-pharyngeal swab taken, mixed in liquid medium, tested in RT-PCR machine
- Can we **pool** (mix) subsets of *n* samples and **test the pools** to save resources?
- Equal portions of participating samples are taken for creating any pool.
- A **negative pool test** implies all contributing samples are **negative (non-infected) .**
- **More work to be done if the pool tests positive (infected).**

Computational problem



Estimate **x** given **y** and **A**

Noise vector

Vector of **observed quantities** proportional to viral loads in *m* pools
m << *n*

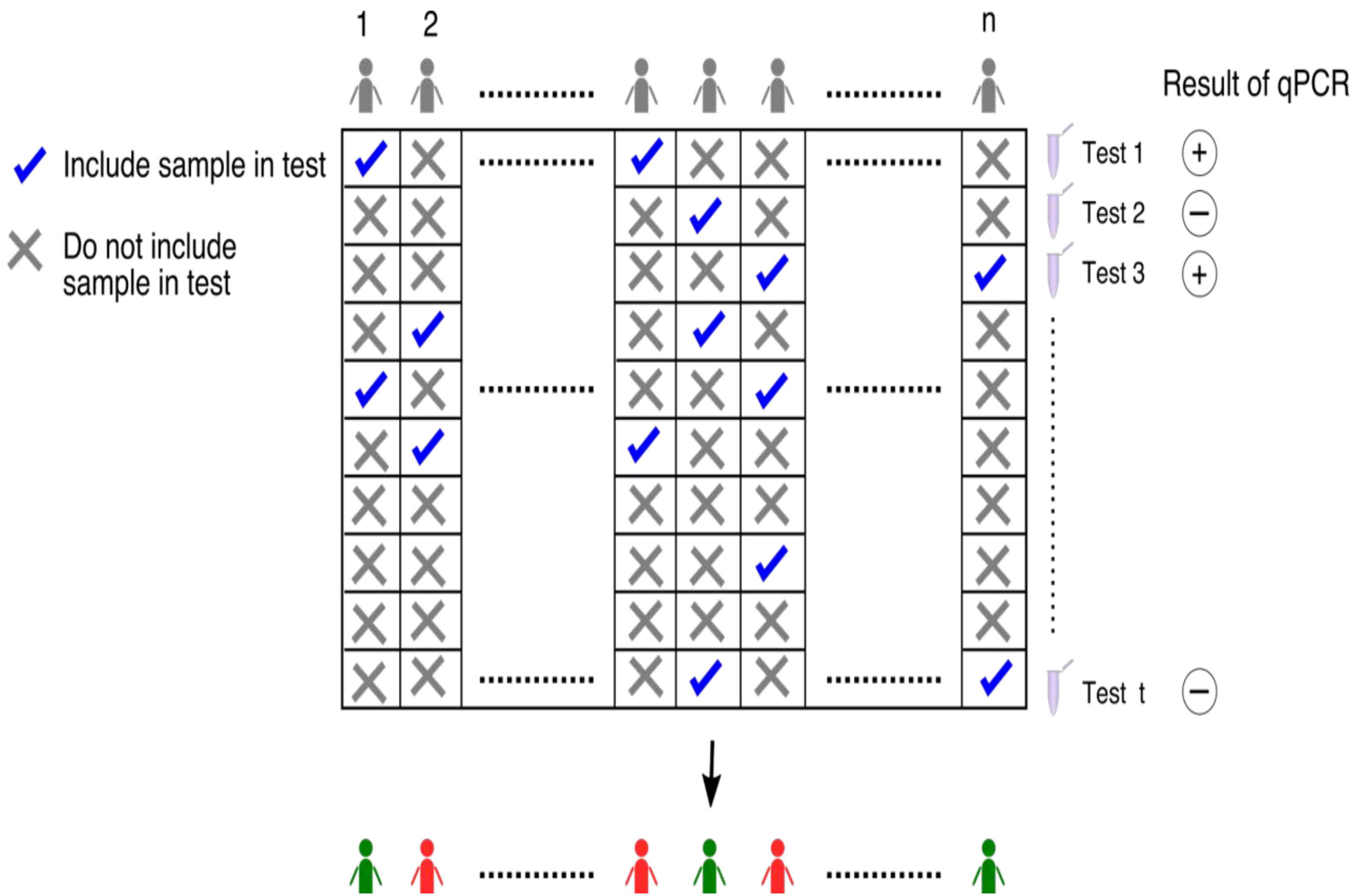
Known mixing matrix:
A_{ij} = 1 if *j*-th sample contributes to *i*-th pool, and 0 otherwise

Sparse vector of **unknown** viral loads in the samples of *n* different people

$$\min \|\boldsymbol{x}\|_1 \text{ s.t. } \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2 \leq \varepsilon.$$

$$J_{lasso}(\boldsymbol{x}; \boldsymbol{y}, \boldsymbol{A}) := \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2 + \lambda \|\boldsymbol{x}\|_1.$$

Designed pooling



What's so interesting about compressive sensing?

- The cool part is that there are provable error bounds between the true **x** and the estimated **x** (i.e. the **x** estimated using a computer algorithm).
- And there are numerous applications.
- So there is a confluence of theory and practice.

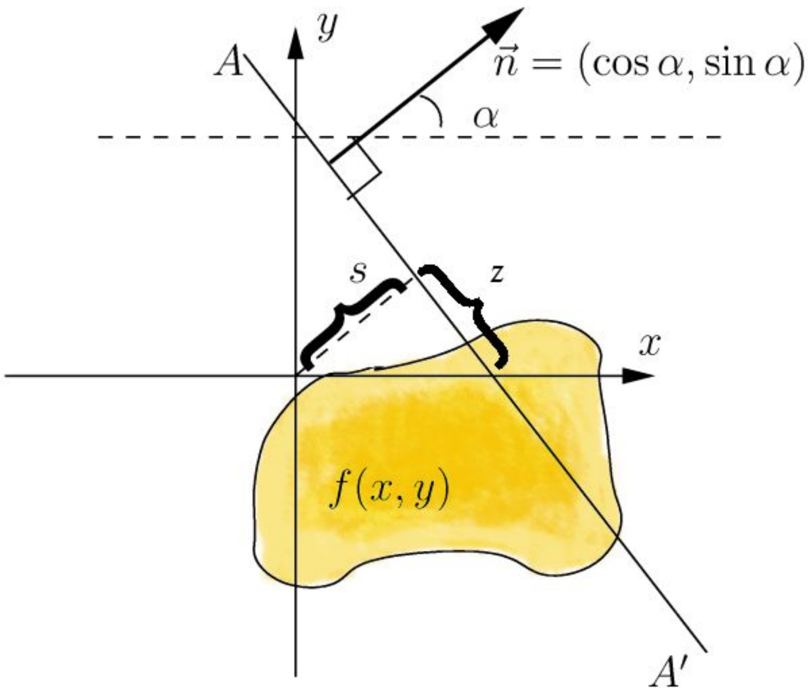
Tomographic reconstruction

- When an X-ray beam is passed through an object **f** at a certain angle, it gets absorbed partially by various materials present inside the object.

- The rest of the X ray beam is collected by a sensor.

- The measurement at the sensor is typically the Radon transform (also called tomogram) of the object – defined as follows:

$$R_\theta(f) = \int f(x,y)dl, l \leftarrow (\cos\theta, \sin\theta)$$



https://en.wikipedia.org/wiki/Radon_transform

Tomographic reconstruction

- The measurement at the sensor is typically the Radon transform of the object – defined as follows:

$$R_\theta(f) = \int f(x,y)dl, l \leftarrow (\cos\theta, \sin\theta)$$

- Such a Radon transform can be computed in different directions θ.
- Each such Radon transform of a 2D object is a 1D signal (or that of a 3D object is a 2D signal).
- The task of reconstructing the 2D object from the given Radon transforms is called **tomographic reconstruction**.

Tomographic reconstruction

- The most popular application of tomographic reconstruction is in medical imaging – CT.
- But there are other applications as well – for example in mechanical engineering, in electron microscopy.
- We will take a look at some of these!

Matrix Completion in Practice: Scenario 1

- Consider a survey of m people where each is asked q questions.
- It may not be possible to ask each person all q questions.
- Consider a matrix of size m by q (each row is the set of questions asked to any given person).
- This matrix is only partially filled (many missing entries).
- **Is it possible to infer the full matrix given just the recorded entries?**

Matrix Completion in Practice: Scenario 2

- Some online shopping sites such as Amazon, Flipkart, Ebay, Netflix etc. have recommender systems.
- These websites collect product ratings from users (especially Netflix).
- Based on user ratings, these websites try to recommend other products/movies to the user that he/she will like with a high probability.
- Consider a matrix with the number of rows equal to the number of users, and number of columns equal to the number of movies/products.
- This matrix will be HIGHLY incomplete (no user has the patience to rate too many movies!!) – maybe only 5% of the entries will be filled up.
- **Can the recommender system infer user preferences from just the defined entries?**

Matrix Completion in Practice: Scenario 2

- Read about the Netflix Prize to design a better recommender system: http://en.wikipedia.org/wiki/Netflix_Prize

Matrix Completion in Practice: Scenario 3

- Consider an image or a video with several pixel values missing.
- This is not uncommon in range imagery or remote sensing applications!
- Consider a matrix whose each column is a (vectorized) patch of m pixels. Let the number of columns be K.
- This m by K matrix will have many missing entries.
- **Is it possible to infer the complete matrix given just the defined pixel values?**
- If the answer were yes, note the implications for image compression!

A property of these matrices

- Scenario 1: Many people will tend to give very similar or identical answers to many survey questions.
- Scenario 2: Many people will have similar preferences for movies (only a few factors affect user choices).
- Scenario 3: Non-local self-similarity!
- This makes the matrices in all these scenarios **(approximately) low in rank!**

Scenario 4: Low rank matrices!

- Consider matrix **D** whose entry D_{ij} = euclidean distance between points $\mathbf{p}_i$ and $\mathbf{p}_j$ in some *k*-dimensional space.
- Such a matrix has rank at the most *k*+2.
- Proof:

$$D_{ij} = \left\| p_i - p_j \right\|^2 = \langle p_i, p_i \rangle + \langle p_j, p_j \rangle - 2\langle p_i, p_j \rangle$$

Column vector

$$\boldsymbol{X} = (p_1 \mid p_2 \mid \ldots \mid p_N), \boldsymbol{X} \in \mathrm{R}^{k \times N}$$

$$\therefore D = diag(\boldsymbol{XX}^T)\mathbf{1}^T + \mathbf{1}diag(\boldsymbol{XX}^T)^T - 2\boldsymbol{XX}^T$$

$$\therefore rank(D) \leq rank(diag(\boldsymbol{XX}^T)\mathbf{1}^T) + rank(\mathbf{1}diag(\boldsymbol{XX}^T)^T) + rank(-2\boldsymbol{XX}^T)$$

$$= 1 + 1 + k$$

https://infoscience.epfl.ch/record/196439/files/EPFL_TH5971.pdf

Low-rank matrices are cool!

- The answer to the four questions/scenarios is a NO in the general case.
- But it's a big YES if we assume that the underlying matrix has low rank (and which, as we have seen, is indeed the case for all four scenarios).
- Low rank matrices of size $n_1 n_2$ with rank r have only $(n_1+n_2-r)r$ degrees of freedom – much less than $n_1 n_2$ when r is small.

Theorem for low-rank matrix completion

- Consider an unknown matrix $\mathbf{M}$ of size n_1 by n_2 having rank $r < \min(n_1, n_2)$.
- Suppose we observe only a fraction of entries of $\mathbf{M}$ in the form of matrix $\boldsymbol{\Gamma}$, where $\boldsymbol{\Gamma}$(i,j) = $\mathbf{M}$(i,j) for all (i,j) belonging to some set Ω and $\boldsymbol{\Gamma}$(i,j) undefined elsewhere.
- If **(1)** $\mathbf{M}$ has row and column spaces that are "sufficiently incoherent" with the canonical basis (i.e. identity matrix), **(2)** r is "sufficiently small", and **(3)** Ω is "sufficiently large", then we can <u>**accurately**</u> recover $\mathbf{M}$ from $\boldsymbol{\Gamma}$ by solving the following **"trace-norm"** minimization problem:

$$(Q0): M^* = \min_M \|M\|_*$$

subject to

$$M(i,j) = \Gamma(i,j) \forall (i,j) \in \Omega$$

Basic Question: Robust PCA

- Consider a matrix **M** of size n1 x n2 that is the sum of two components – **L** (a low-rank components) and **S** (a component with sparse but unknown support).
- Can we recover **L** and **S** given only **M**?
- Note: support of **S** means all those matrix entries (i,j) where S(i,j) $\neq$ 0.

Why ask this question? Scenario 1

- Consider a video taken with a static camera (like at airports or on overhead bridges on expressways).
- Such videos contain a “background” portion that is static or changes infrequently, and a moving portion called “foreground” (people, cars etc.).
- In many applications (esp. surveillance for security or traffic monitoring), we need to detect and track the moving foreground.

Why ask this question? Scenario 1

- Let us represent this video as a N x T matrix, where the number of columns T equals the number of frames in the video, and where each column containing a single video-frame of $N = N_1 N_2$ pixels converted to a vector form.
- This matrix can clearly be expressed as the sum of a low-rank background matrix and a sparse foreground matrix.

Theorem for Robust PCA

- Consider a matrix **M** of size n_1 by n_2 which is the sum of a "sufficiently low-rank" component **L** and a "sufficiently sparse" component **S** whose support is uniformly randomly distributed in the entries of **M**.
- Then the solution of the following optimization problem (known as **principal component pursuit**) yields **exact estimates** of **L** and **S** with "very high" probability:

$$E(L',S') = \min_{(L,S)} \|L\|_* + \frac{1}{\sqrt{\max(n_1,n_2)}} \|S\|_1$$

subject to $L + S = M$.

$$Note: \|S\|_1 = \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} |S_{ij}|$$

This is a convex optimization problem.

Logistics/Administrative Stuff

Mathematical Tools

- Numerical linear algebra: eigenvectors and eigenvalues, SVD, matrix inverse and pseudo-inverse – **you are expected to know this (but if not, I will help).**
- Signal processing concepts: Fourier transform, convolution, discrete cosine transform – **you are expected to know this (but if not, I will help).**
- Some machine learning methods (will be covered in class)

Programming tools

- **MATLAB and associated toolboxes**
- OpenCV (open source C++ library)

Logistics

- Associated slides will be shared 2 days in advance
- Lectures will be conducted every Tuesday and Friday from 5:30 pm to 7:00 pm (slot 14)
- Venue: CC 103
- Office hours immediately **after** class (7:00 to 7:30 pm)
- Grading scheme, audit policy, attendance policy, etc. on course webpage
- We will make extensive use of moodle for sharing of class material.