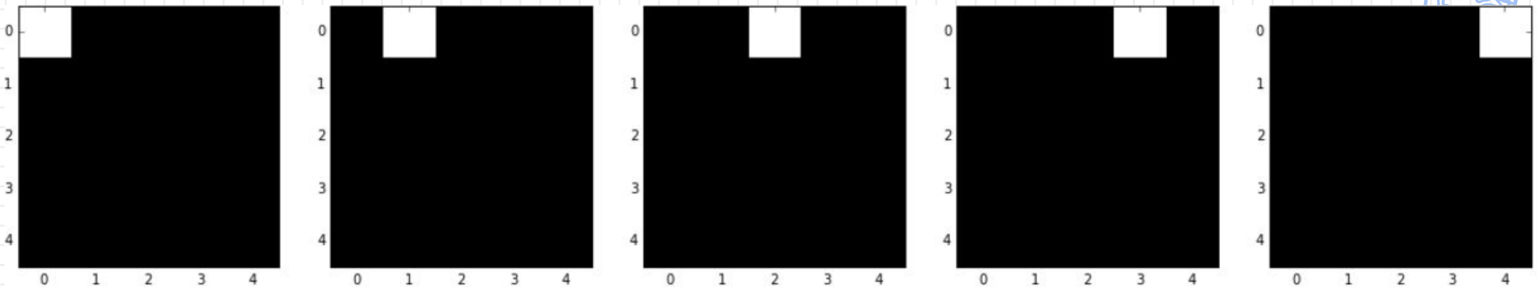
EECS16A Imaging 3

Announcements

- Midterm I is this week, Friday 7/10

- Buffer labs are next week, 7/13 & 7/14
  - You can make up **one** missed lab from the Imaging Module, if needed
  - Held during your regular lab section (this one)
  - Touchscreen 1 starts on Wednesday 7/15

Last time: Single-pixel scanning



- Setup a masking matrix where each row is a mask
  - Measured each pixel individually once

Last time: Matrix-vector multiplication

1	0	0	0	0	0	0	0	...
0	1	0	0	0	0	0	0	...
0	0	1	0	0	0	0	0	...
0	0	0	1	0	0	0	0	...
0	0	0	0	1	0	0	0	...
0	0	0	0	0	1	0	0	...
0	0	0	0	0	0	1	0	...
...								

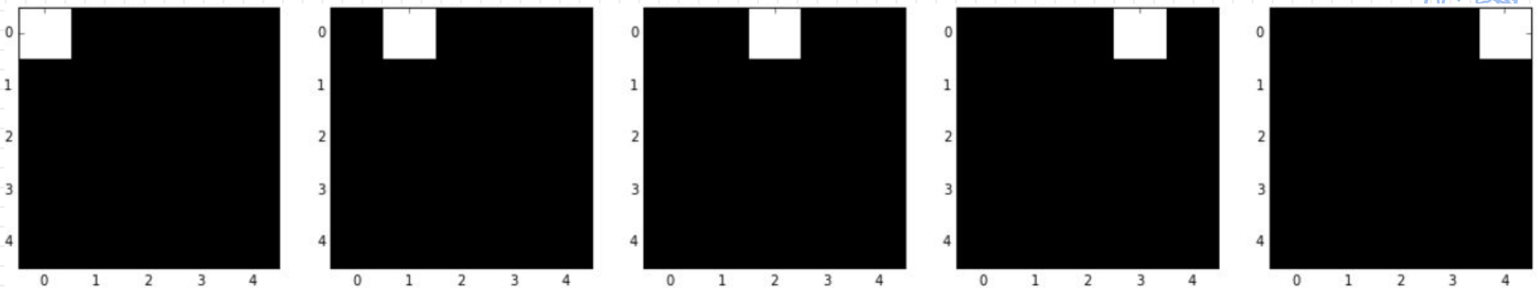
$$\begin{bmatrix} i_1 \\ i_2 \\ i_3 \\ \\ \\ \\ i_n \end{bmatrix} = \begin{bmatrix} s_1 \\ s_2 \\ s_3 \\ \\ \\ \\ s_n \end{bmatrix}$$

Masking Matrix H

Unknown, vectorized image, $\vec{i}$

Recorded Sensor readings, $\vec{s}$

Last time: Single-pixel scanning



- Setup a masking matrix where each row is a mask
    - Measured each pixel individually once
- **How did we reconstruct our image, once we had s?**
- **What are the requirements of our masking matrix H?**

$$\vec{s} = H\vec{i}$$

Questions from Imaging 2

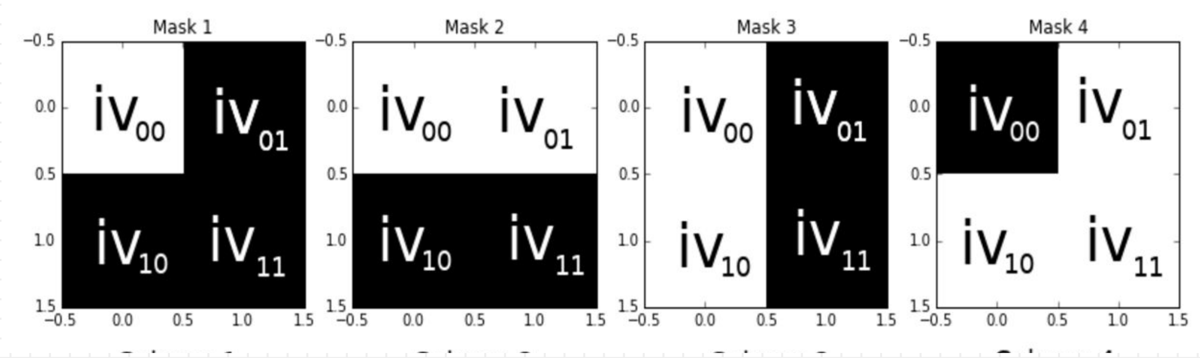
Goal: Understand which measurements are good measurements

✓ Can we always reconstruct our image → **need invertible H**

? Are all invertible matrices equally good as scanning matrices?

? What happens if we mess up a single scan?

Today: Multi-pixel scanning



- **Can we measure multiple pixels at a time?**
    - Measurements are now linear combinations of pixels

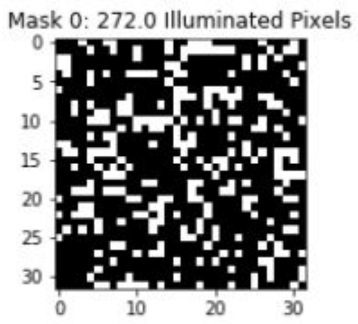
- **How can we reconstruct our scanned image?**

Why do we care?

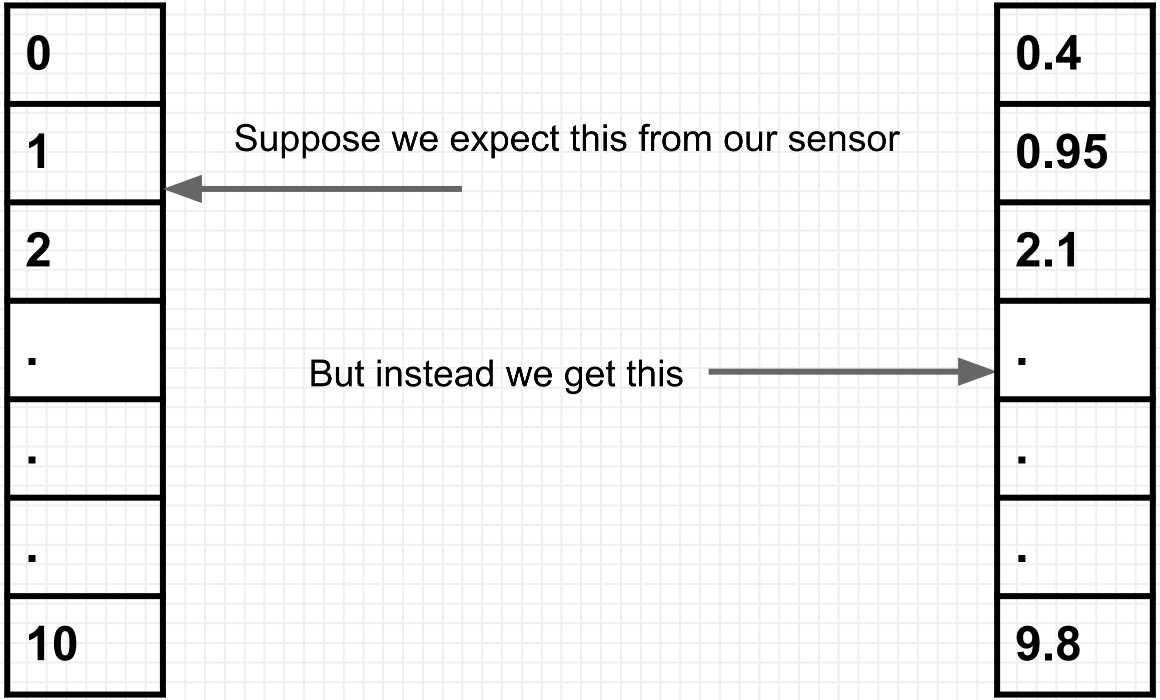
- Improve image quality by averaging
  - Good measurements → good average
- Redundancy is useful
  - Averaging measurements is better than using bad measurement values
  - Does not “solve” bad measurements, but makes us tolerant of some errors

How do we do it?

- Change masks to illuminate multiple pixels per scan
  - Multiple 1’s in each row of masking matrix H
  - Measure linear combinations of pixels instead of single pixels
- BUT multiple pixels → more noise
  - Noise = random variation in our measurement that we don’t want (ex: room light getting into box)
  - Signal = data that we do want (light from pixel illumination)
- Too much noise → hard to distinguish signal from noise
  - Want high signal, low noise
  - High signal-to-noise ratio (SNR)



What is noise?



What is Noise?

Measured values =
ideal vector + noise vector (ω)

$$\begin{bmatrix} 0.4 \\ 0.95 \\ 2.1 \\ . \\ . \\ . \\ 9.8 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 2 \\ . \\ . \\ . \\ 10 \end{bmatrix} + \begin{bmatrix} 0.4 \\ -0.05 \\ 0.1 \\ . \\ . \\ . \\ -0.2 \end{bmatrix}$$

$$\vec{S}_{real} \qquad \vec{S}_{ideal} \qquad \vec{w}$$

How does noise affect our system?

$$
\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \dots \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & \dots \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & \dots \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & \dots \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & \dots \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & \dots \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & \dots \\
 & & & & \dots & & & &
\end{bmatrix}
\begin{bmatrix} i_1 \\ i_2 \\ i_3 \\ \\ \\ \\ i_n \end{bmatrix}
+
\begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ \\ \\ \\ \omega_n \end{bmatrix}
=
\begin{bmatrix} s_1 \\ s_2 \\ s_3 \\ \\ \\ \\ s_n \end{bmatrix}
$$

Masking Matrix **H**

Unknown, vectorized image, $\vec{i}$

Random noise vector, $\vec{w}$

Recorded Sensor readings, $\vec{s}$

A more realistic system

- Sensor readings =
  image vectors applied to H + noise vector

$$\vec{s} = H\vec{i} + \vec{w}$$

- We can't reconstruct **i**, but we can estimate it

$$\vec{i}_{est} = H^{-1}\vec{s} = \vec{i} + \boxed{H^{-1}\vec{w}}$$

Be careful about the noise term or else it could blow up !!

Eigenvalues for inverse matrices

- H Is an NxN matrix that we know is linearly independent (invertible).
  - No eigenvalue = 0
- Assume H has N linearly independent eigenvectors
- $\mathrm{H}v_i = \lambda_i v_i \text{ for } i = 1 \ldots \mathrm{N}$
- N lin. ind. vectors can span $\mathbb{R}^N$
  - They span the noise vector

- The inverse of H has eigenvalues $\frac{1}{\lambda_1} \ldots \frac{1}{\lambda_N}$ (as proven in homework)

$$\mathrm{H}^{-1}v_i = \frac{1}{\lambda_i}v_i \text{ for } i = 1 \ldots \mathrm{N}$$

How do eigenvalues affect noise?

The noise vector can be written as:

$$\vec{\omega} = \alpha_1 \vec{v_1} + \alpha_2 \vec{v_2} + \cdots \alpha_n \vec{v_n}$$

Including effect of H^{-1}

$$H^{-1}\vec{\omega} = H^{-1}(\alpha_1 \vec{v_1} + \alpha_2 \vec{v_2} + \cdots \alpha_n \vec{v_n})$$

Rewritten with eigenvalues:

$$H^{-1}\vec{\omega} = \frac{1}{\lambda_1}\alpha_1 \vec{v_1} + \frac{1}{\lambda_2}\alpha_2 \vec{v_2} + \cdots \frac{1}{\lambda_n}\alpha_n \vec{v_n}$$

Linking it all together

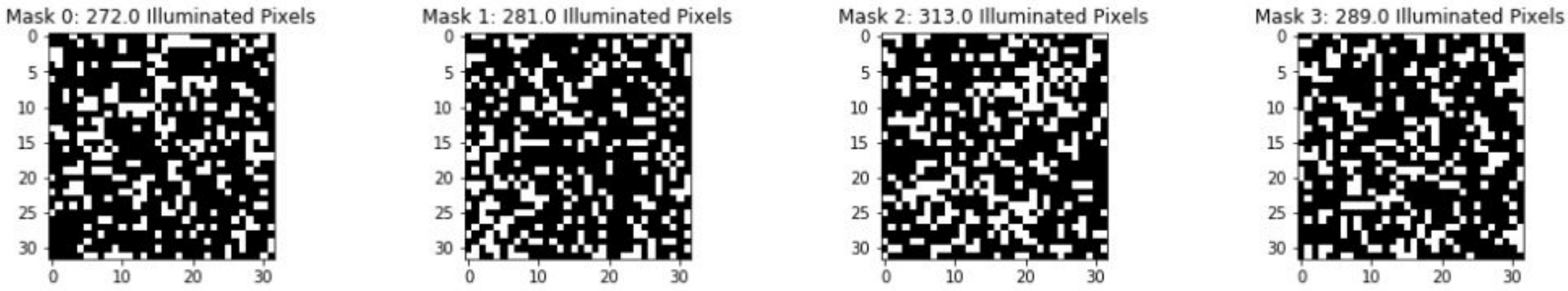
$$\vec{i}_{est} = H^{-1}\vec{s} + \boxed{H^{-1}\vec{\omega}}$$

$$\boxed{H^{-1}\vec{\omega}} = \frac{1}{\lambda_1}\alpha_1\overrightarrow{v_1} + \frac{1}{\lambda_2}\alpha_2\overrightarrow{v_2} + \cdots \frac{1}{\lambda_n}\alpha_n\overrightarrow{v_n}$$

- Remember: want small noise term for high signal-to-noise ratio
- The noise is directly related to the eigenvalues.

- **Do we want small or large eigenvalues for the H matrix in order to get a good image?**

Possible Scanning Matrix: Random

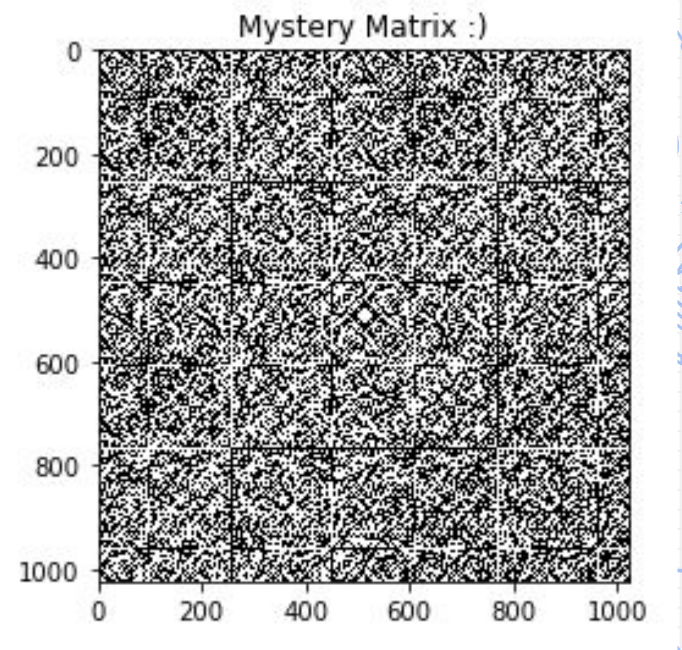


- Illuminate ~300 pixels per scan
  - *Usually* invertible
  - But what are its eigenvalues?

¯_(ツ)_/¯

A more systematic scanning matrix:

- Hadamard matrix!
- Constructed to have large eigenvalues
  - Just what we need!



Software Simulator Setup

1. Upload ("simple") image of an object (or use 16A logo)
2. Convert image to 32x32 form for the simulator
3. Project masks (rows of H) onto it and "measure" **s** using matrix multiplication
4. Multiply with H inverse to find **i** ($=H^{-1}\mathbf{s}$)

The simulator handles noise addition in step 3 using a parameter sigma

Using the Software Simulator

1. Start display view in another browser tab
2. Enter the imagePath and run the simulator + shift to display tab
3. Observe masks being projected onto the image + return to notebook tab
4. Observe generated sensor reading
5. Reconstruct image by multiplying with H inverse

Repeat steps 2-5 for each imaging experiment

Pointers

1. READ CAREFULLY - Long lab with lots of reading; heavily tests understanding of eigen-stuff (important for the exam)
2. You see the noisy sensor reading generated at the end instead of being generated entry by entry (i.e. just one masking simulation visual per experiment, no more cumulative simulation)
3. Choose an image that focuses on a single object and is not too detailed
4. Use a simple imagePath name
5. Before starting the imaging experiments, launch the display view in a separate tab using the link in the notebook
6. Enter imagePath correctly for each simulation block
7. Shift to the display tab as soon as you run a simulation block and return to the notebook once the visual has finished executing

P.S.