CS463/516

Lecture 8

Image registration

- Two images S and C. want transform T such that $T(S)$ resembles C

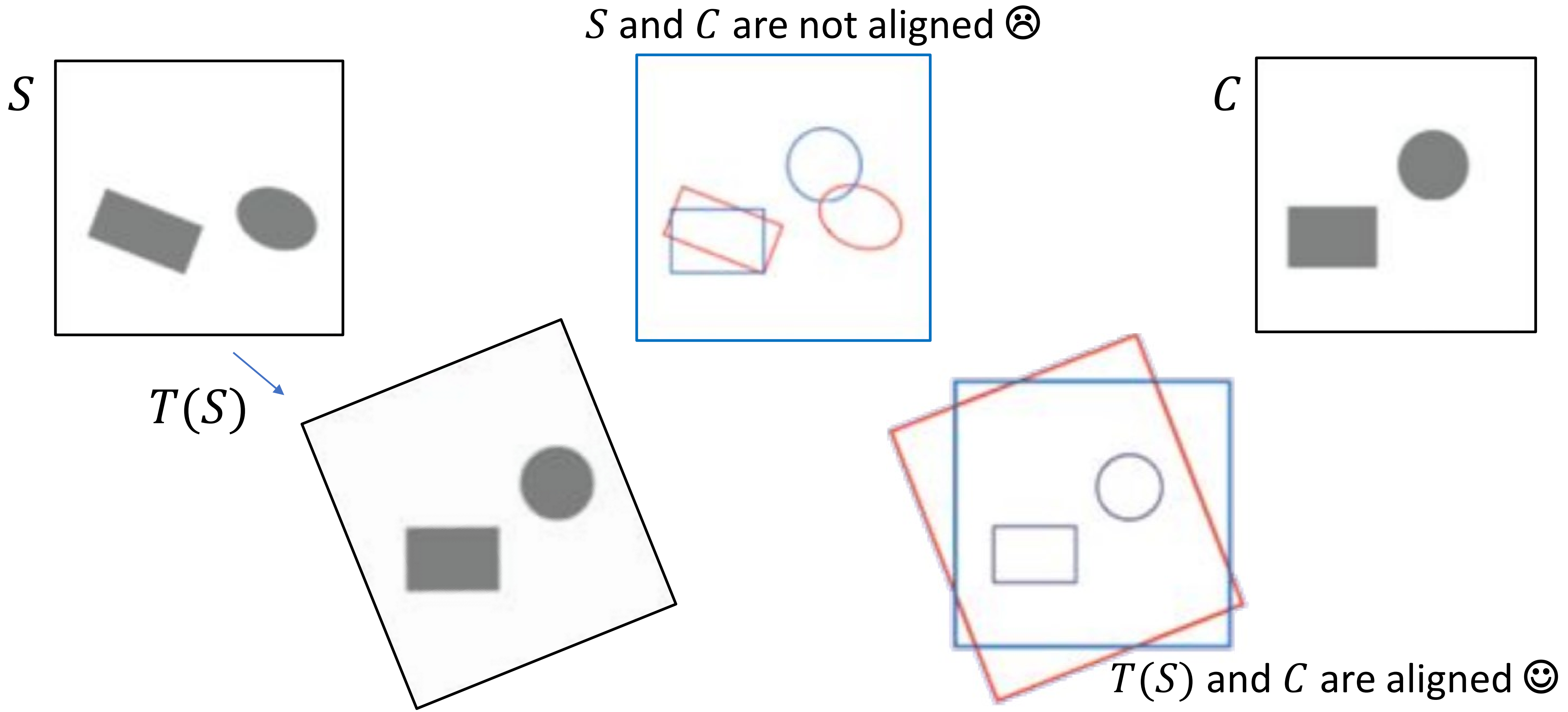
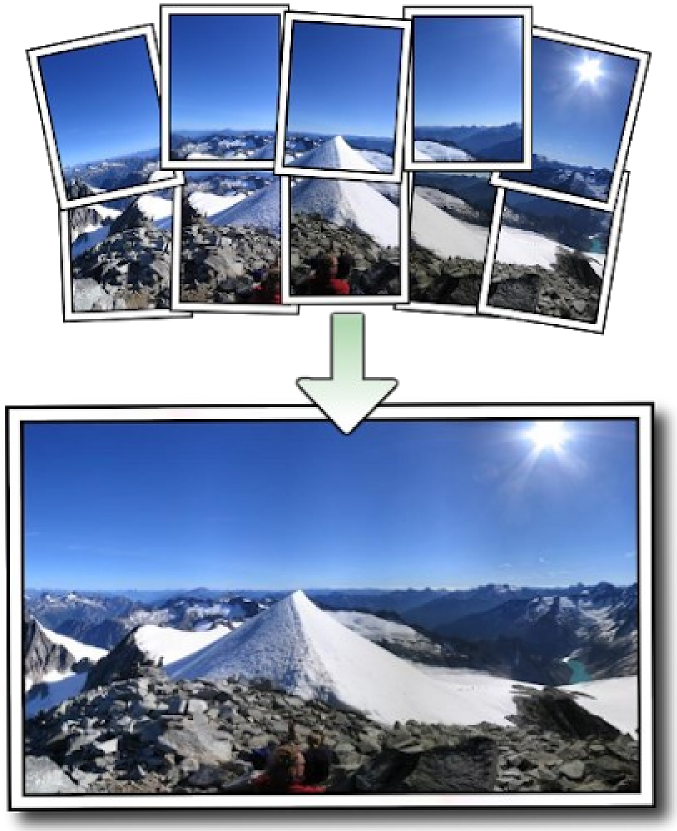


Image registration applications

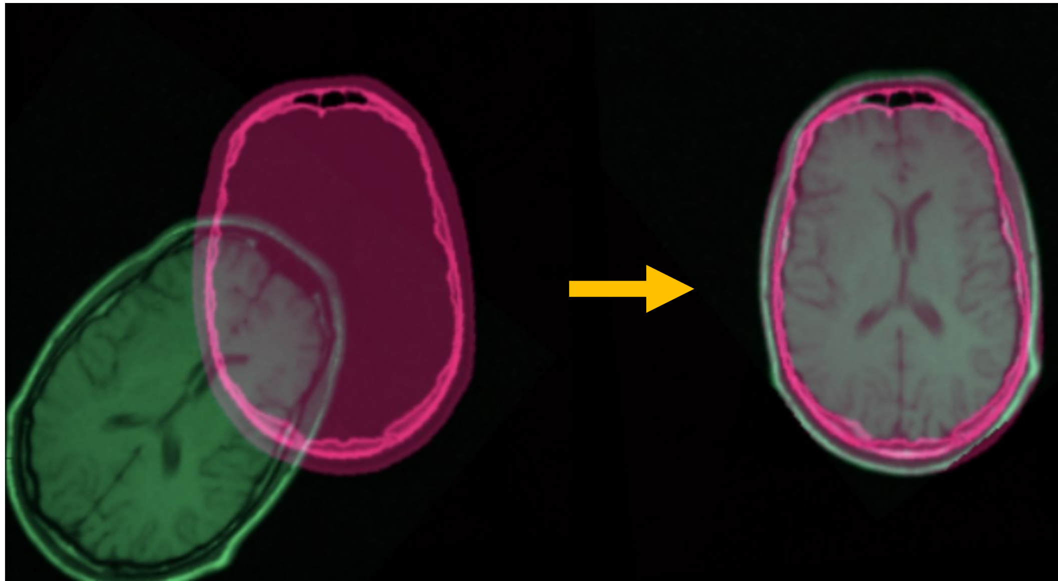
- a) Multi-modal analysis (medical imaging)
    - Align a CT and an MRI image
- b) Image stitching
    - Create large, high-resolution image by registering multiple smaller images together
- c) Panorama creation
    - Align images taken at different angles

b)



c)

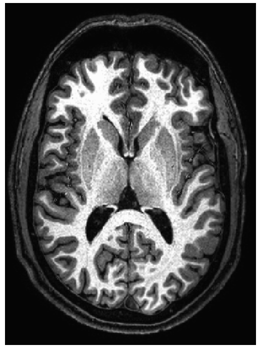
a)



Application: atlas-based analysis

- Atlas-based analysis
  - Atlas of brain regions defined on a template brain
    - This type of atlas typically created by anatomy expert (Doctor or researcher)
  - How to use the atlas on other subjects, not just the template?
  - Solution: use image registration to align template with single subjects
    - Can then apply the transformation to atlas, aligning atlas with each subject

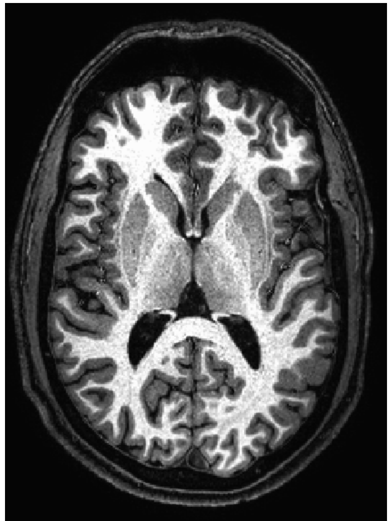
align



template brain

Atlas overlay on template

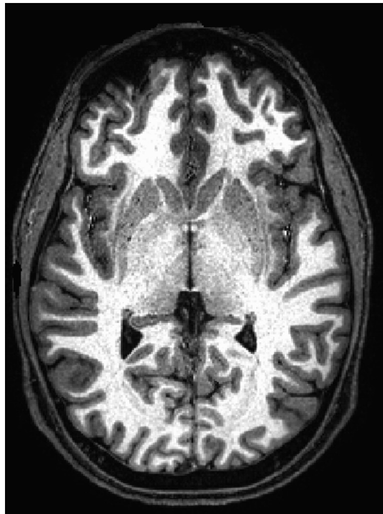
Subject 1



Subject 2

…

Subject N

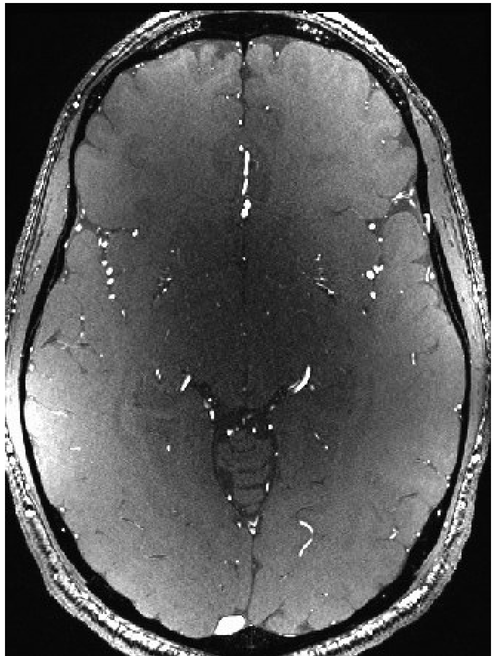


Application: multi-modal alignment

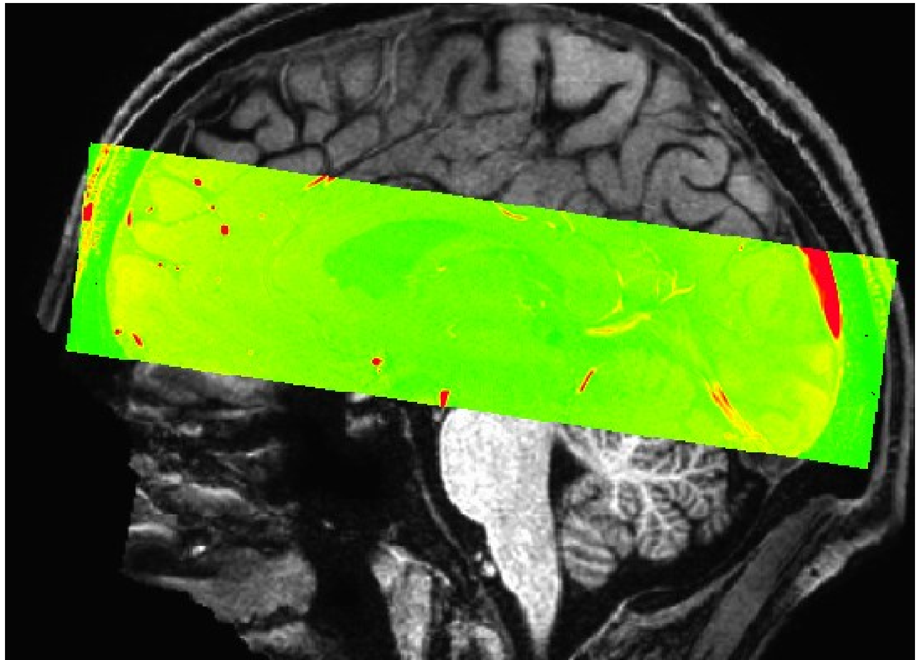
- In MRI (and other methods), often acquire multiple images with different contrasts during same experiment
- Would like to align these images so can examine them all in 'same space'
- Example: T1 (a) and TOF (b), both acquired in same subject
  - However, subject may have moved between scans.
  - Also, resolution and slice positioning is different, TOF is higher resolution so we only acquire a thin 'slab' of slices
- Solution: register the T1 to the TOF (or vice versa) (c)
- Now, we can combine the modalities for more interesting and powerful analysis

a) T1

b) TOF

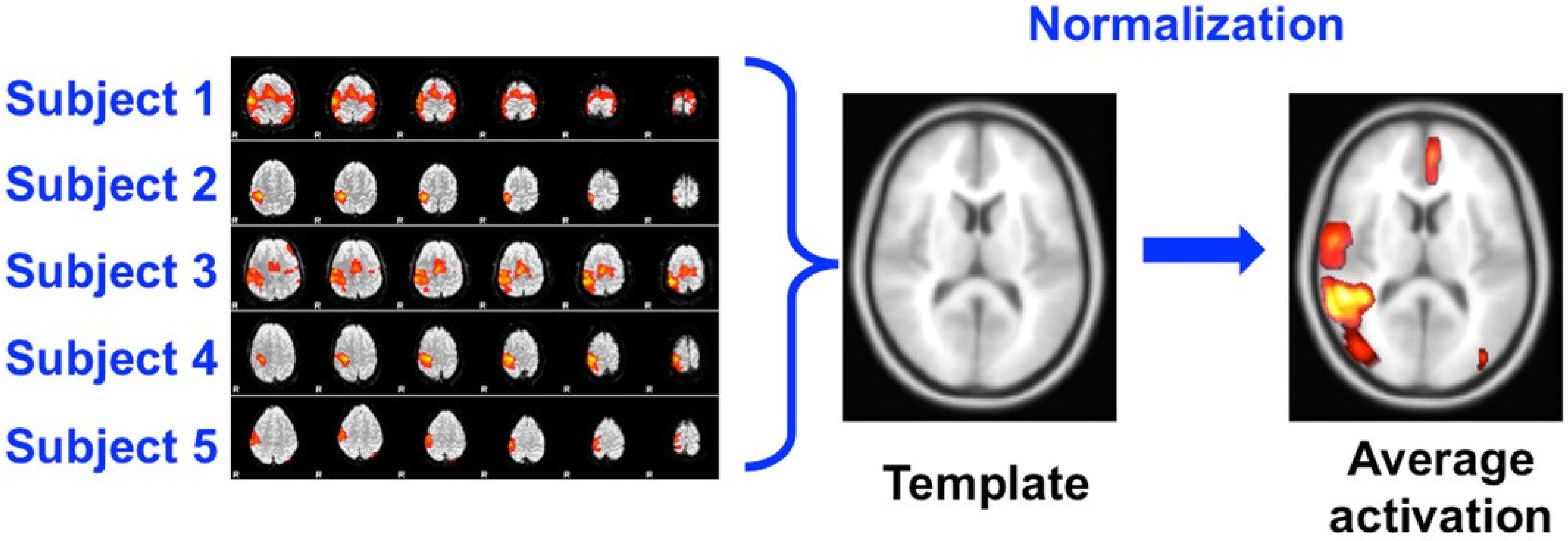


c) TOF (green/red)overlaid on T1



Application: inter-subject averaging

- Common to perform same experiment on multiple subjects then average the results across subjects, to check for a statistically significant effect
- Problem: different people have differently shaped brains!
- Need all subjects to be in same space when we average
  - Otherwise information is lost because signal gets cancelled out due to mis-alignment
- Example: fMRI task on 5 different subjects. Align all subjects with template and *then* average

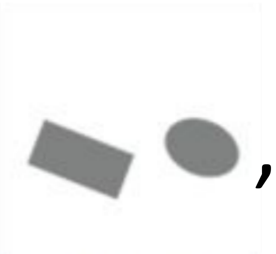


Principles of image registration

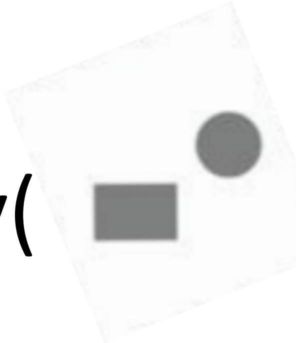
- **Need 3 things to successfully align two images:**
- 1) similarity criterion
  - Measure of ‘how different’ are two images
  - Similar to cost function from machine learning
- 2) transformation method
  - Typically, linear (affine) transform is used, has 12 ‘degrees of freedom’
  - Methods for nonlinear transformation also exist
- 3) optimization method
  - Gradient descent

Similarity criterion

- How to know if two images are similar?
- Define some hypothetical function ‘Similarity’:

Similarity( , ) = ☹

I J

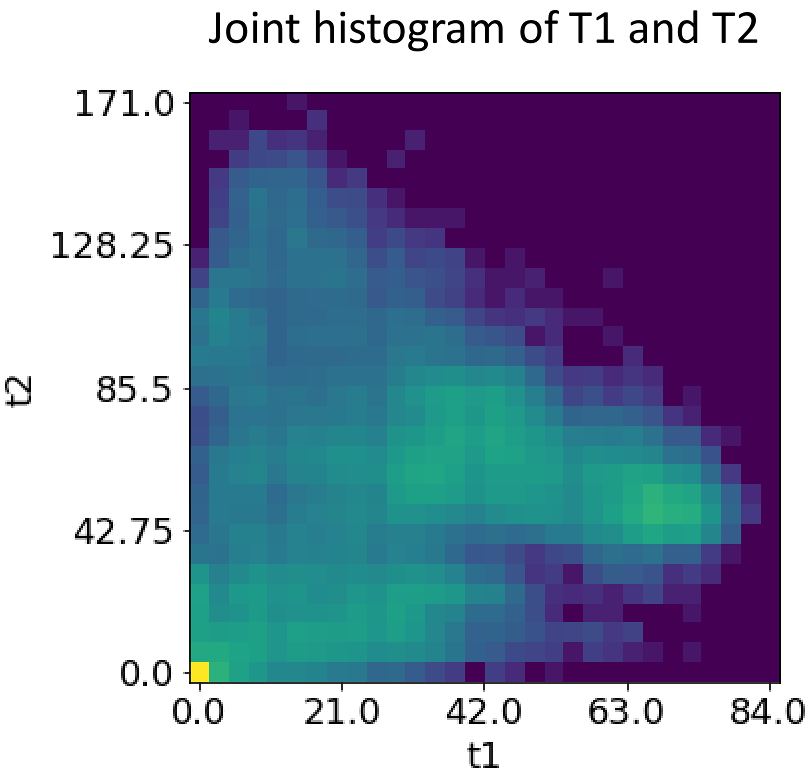
Similarity( , ) = ☺

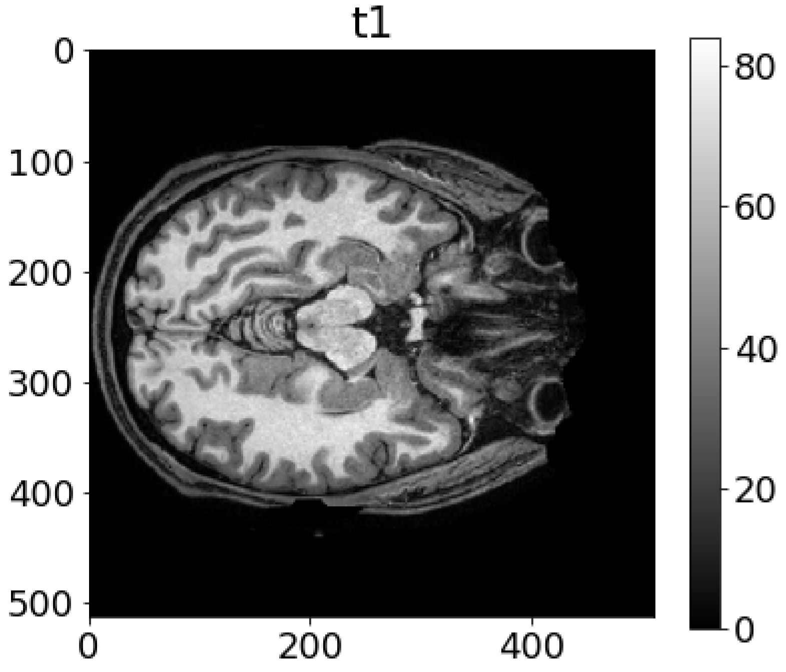
$T(I)$ J

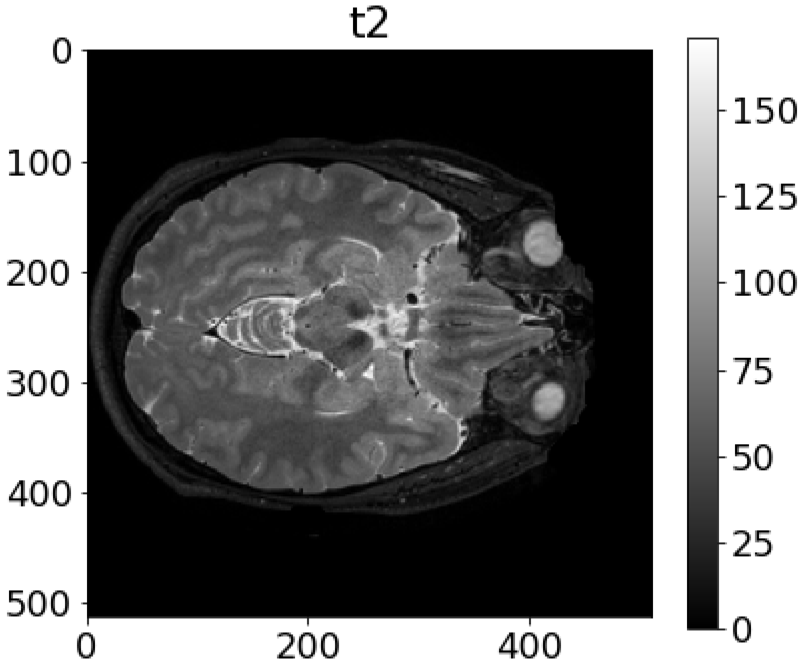
- Also, choose a family of transforms F
- Now, can write the image registration problem as:
- $argmin_{T \in F}\{\, Similarity(\mathrm{T(I), J})\, \} = ?$
  - In other words, want to find the transformation T that, when applied to image I gives us the most similarity to image J

Similarity criteria

- Any ideas?
- Idea 1: joint histogram of the two images
- **Joint histogram:**
- Shows number of times that value x in T1 and value y in T2 occur in same place (overlap)
- $H_{I,J}(i,j) = Card\{(x,y) \mid I(x,y) = i \ and \ J(x,y) = j\}$



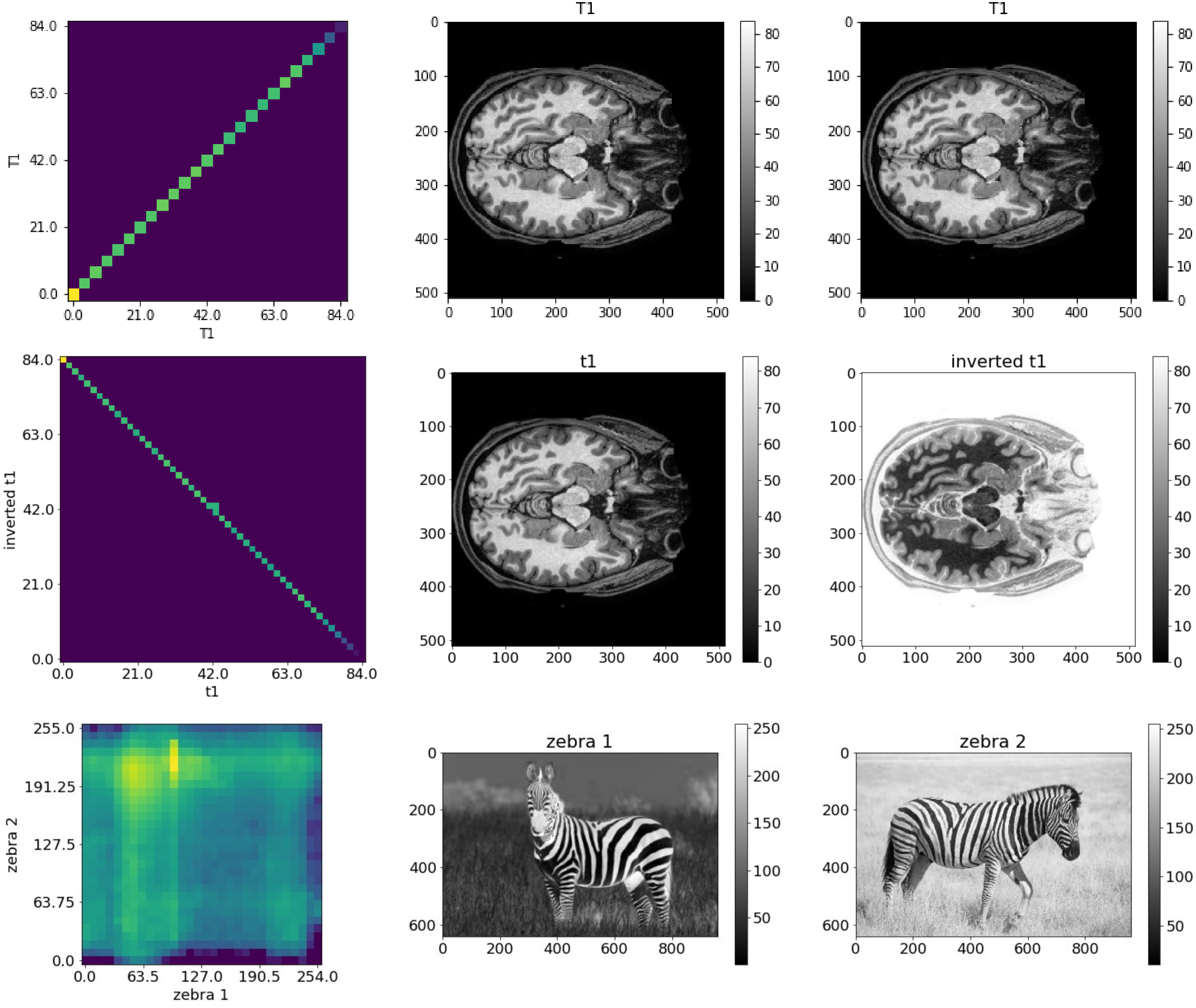




Joint histogram

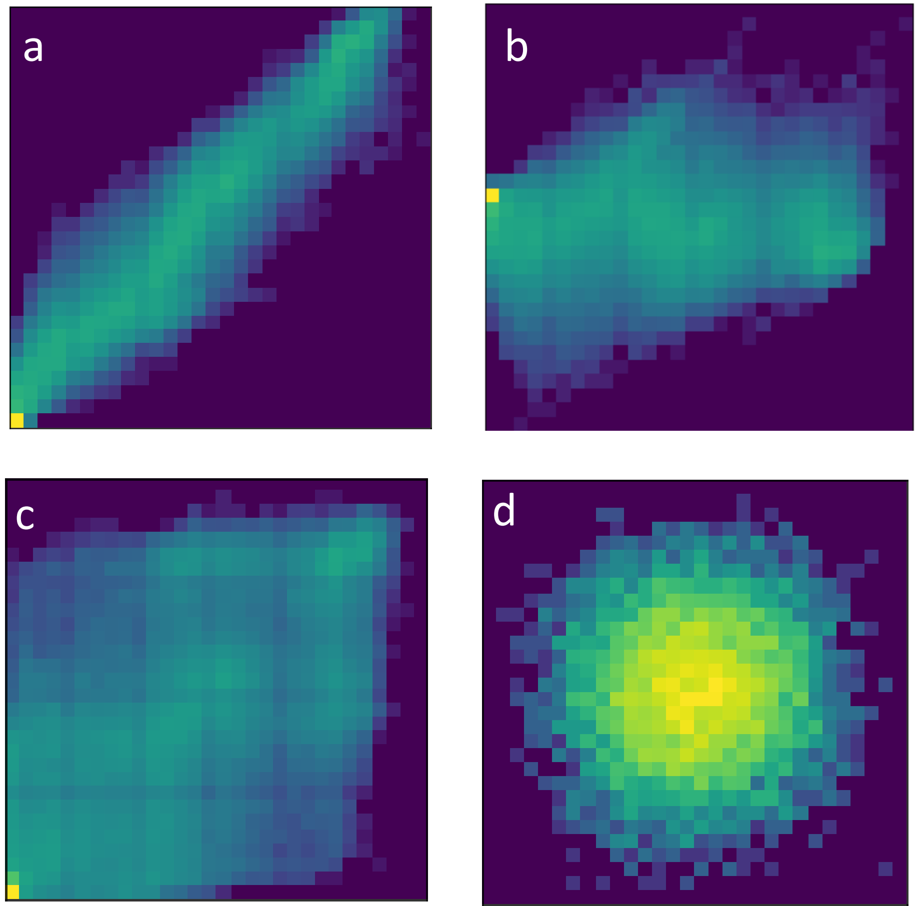
$$H_{I,J}(i,j) = Card\{(x,y) \mid I(x,y) = i \text{ and } J(x,y) = j\}$$

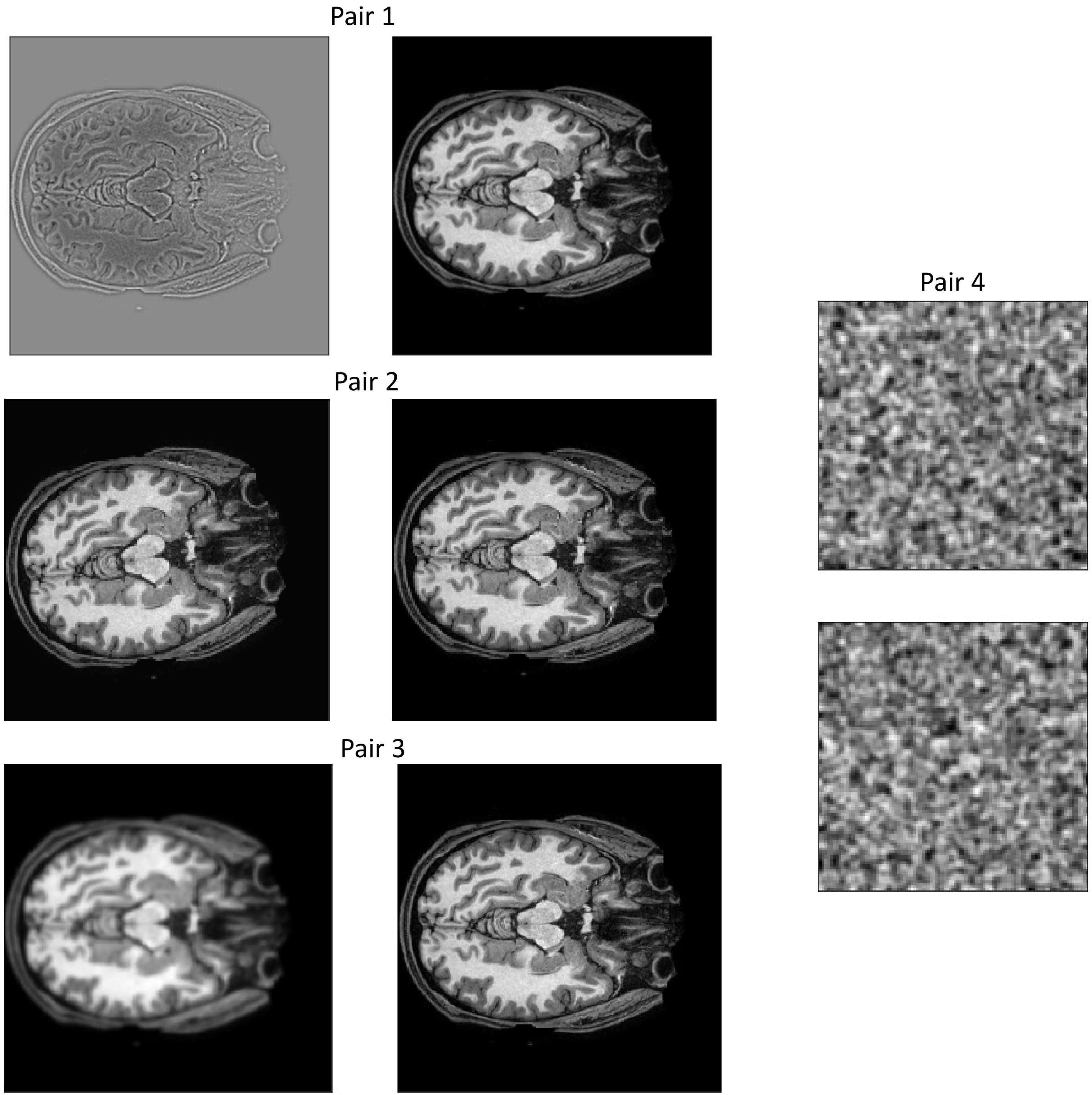
- More examples:
- Same image
    - n_bins = 25
- Inverted image
    - n_bins = 50
- zebra



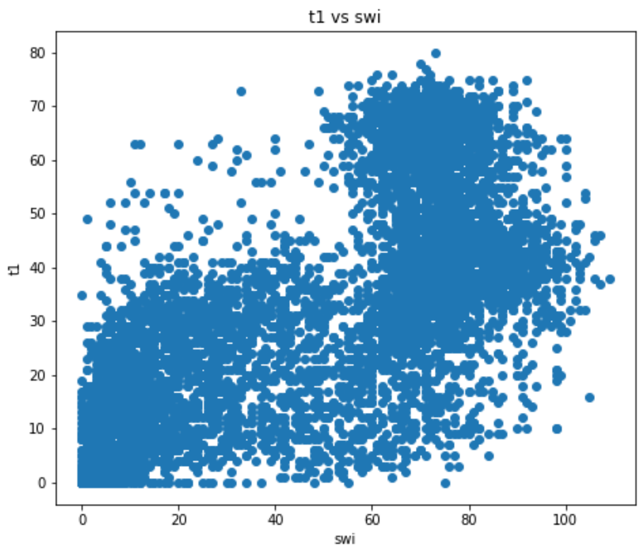
Joint histogram quiz

- Match the image pairs (1-4) to their joint histogram (a-d)



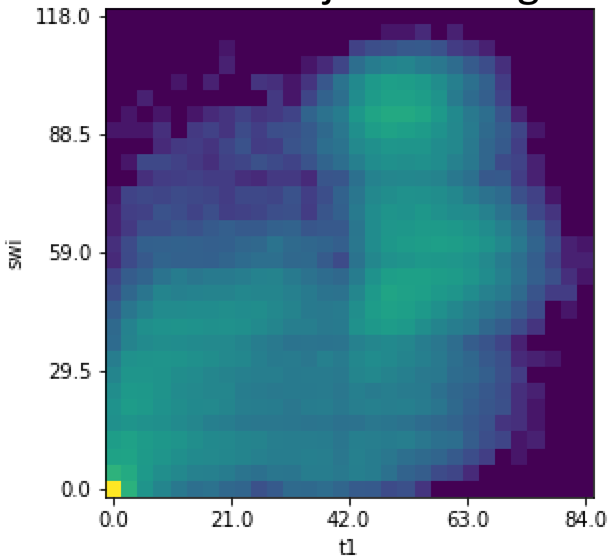


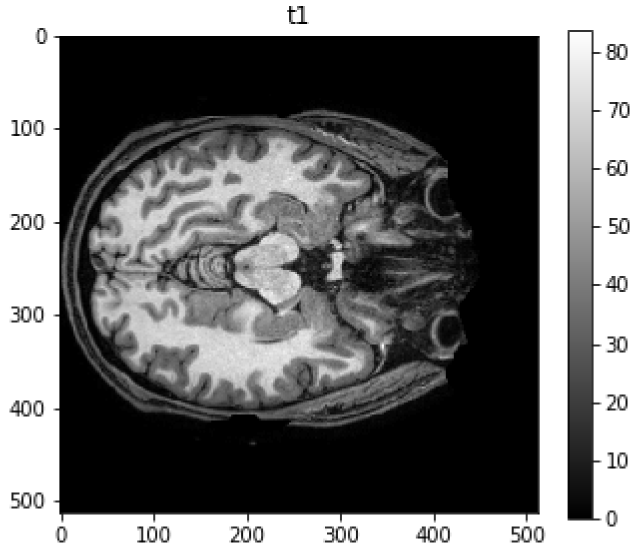
Joint histogram, scatter plot

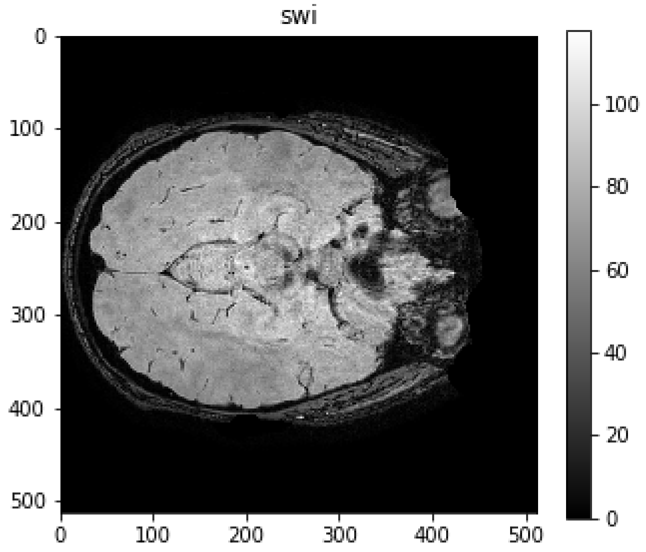


- can think of joint histogram as 'heat map' of scatter plot
- Example: swi and t1 joint histogram (a)
- Swi and t1 scatter plot (b)

```
plt.plot(np.ravel(swi[:,:,200]),np.ravel(t1[:,:,200]),'o');
```





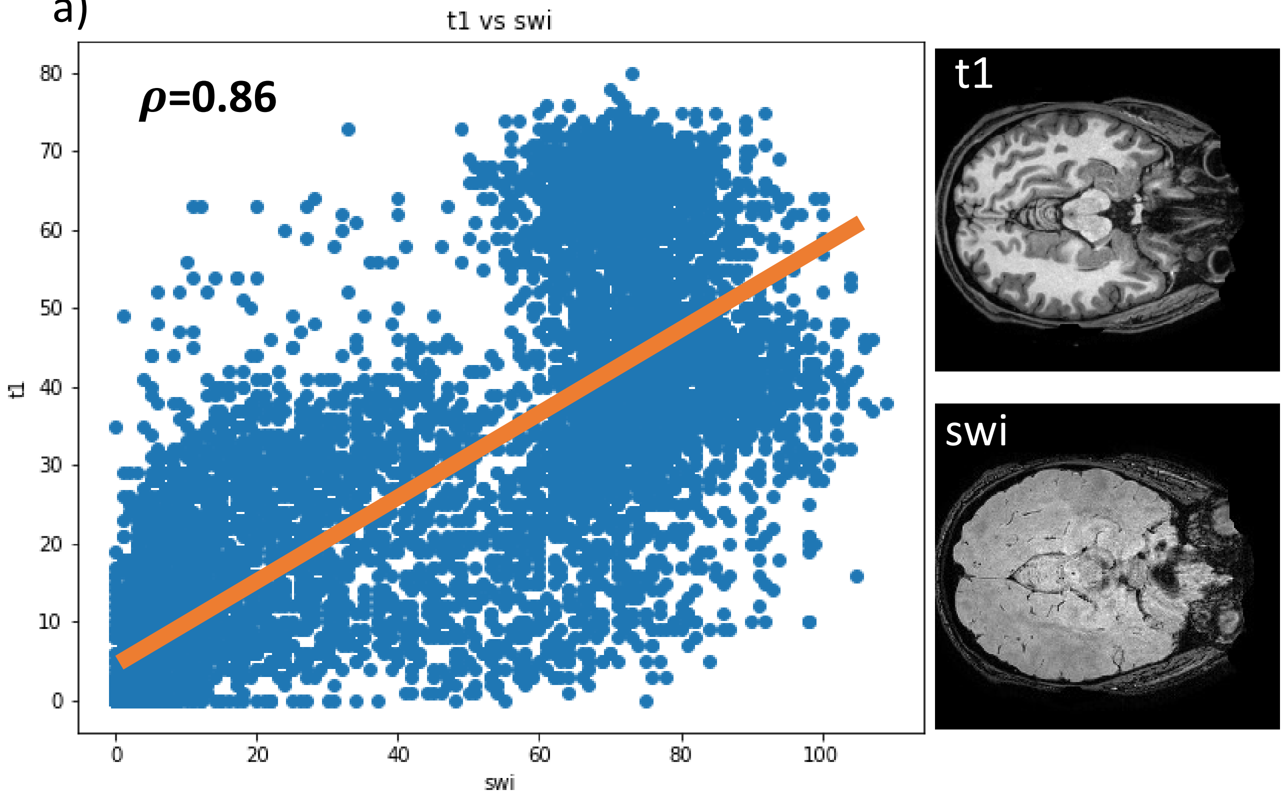


Similarity criterion: correlation coefficient (ρ)

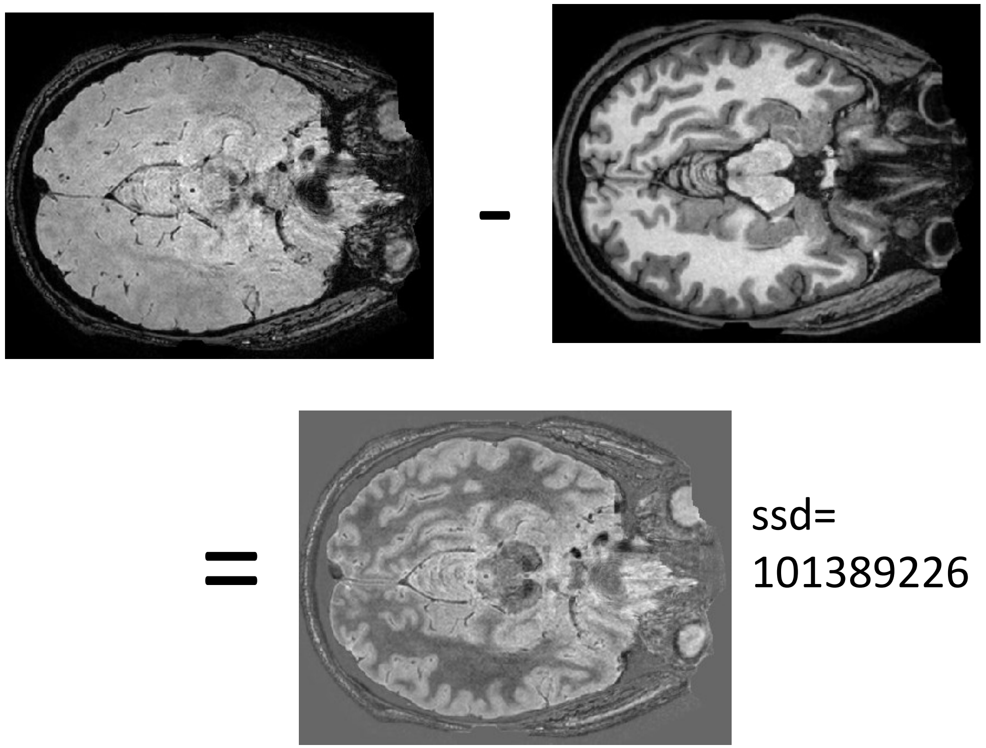
- Can use correlation coefficient to quantify how similar are two images (a)
- Why is correlation coefficient so high (images don't look so similar)
- Another metric: sum-squared difference (b) (not used very often)
  - Can be used in some quantitative applications, intra-subject, or on binary images

$$\rho(I,J) = \frac{\sum_{x,y}(I(x,y)-\bar{I})(J(x,y)-\bar{J})}{\sqrt{\sum_{x,y}(I(x,y)-\bar{I})^2}\sqrt{\sum_{x,y}(J(x,y)-\bar{J})^2}}$$

a)

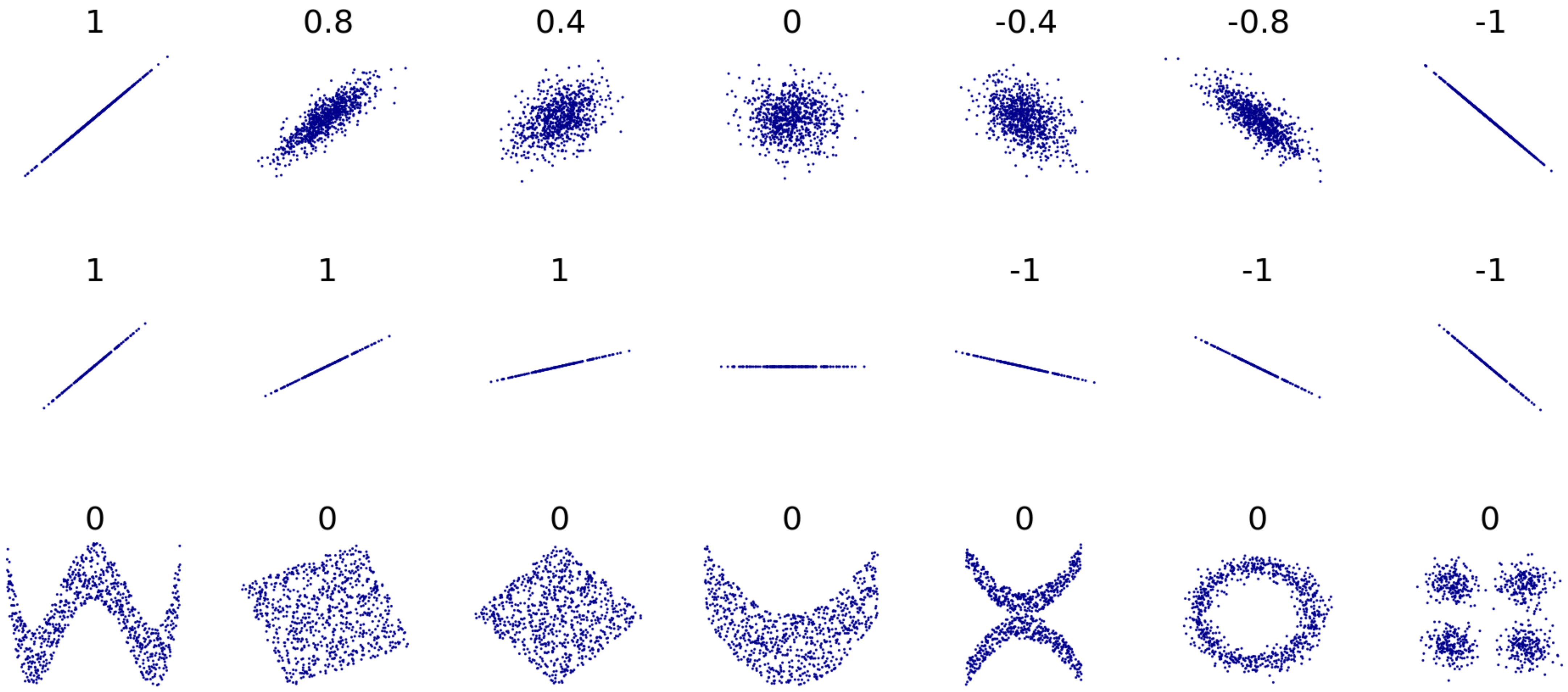


b)



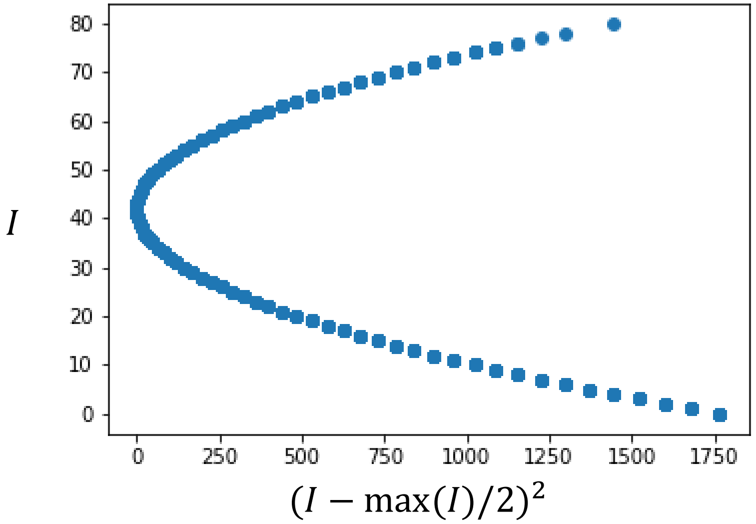
Correlation coefficient ρ

- ρ not a perfect measure of functional dependence (how y depends on x)

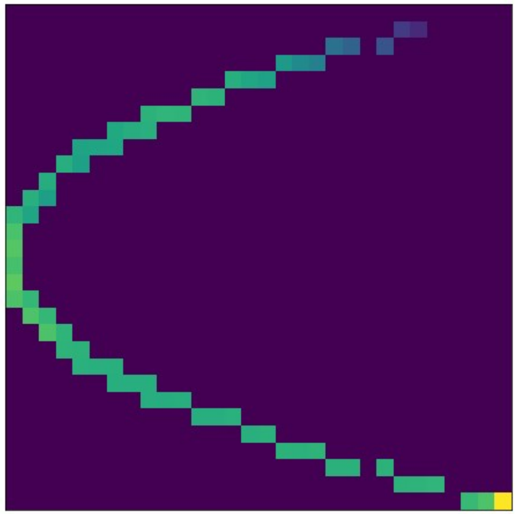


When correlation coefficient ρ fails

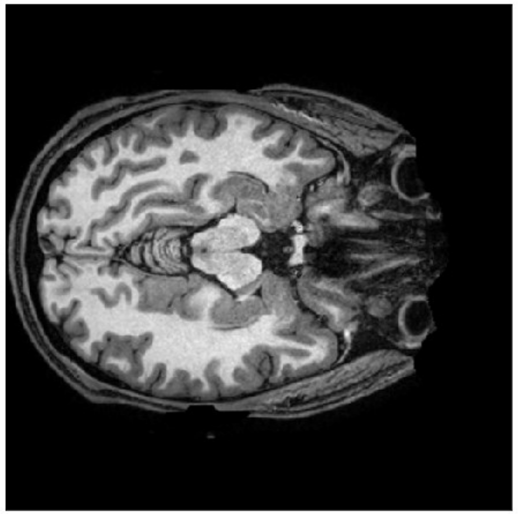
- Reminder: want measure that tell us when $T(I)$ is close to J
- Often, I and J are related in a way that ρ fails to quantify correctly
  - In these cases registration will fail
- Need a better, more flexible similarity criteria (neither ρ or ssd work well)



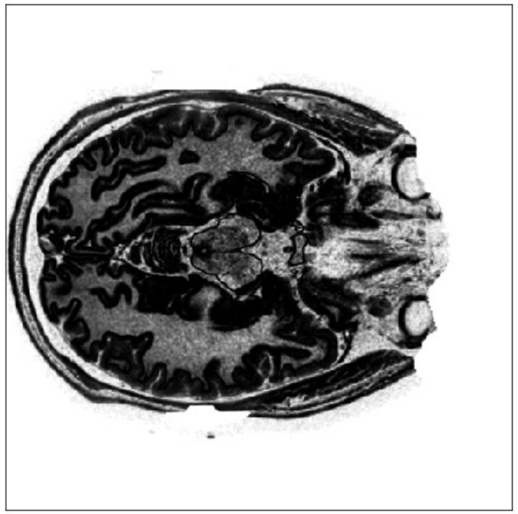
Joint histogram



I



$J = (I - \max(I)/2)^2$

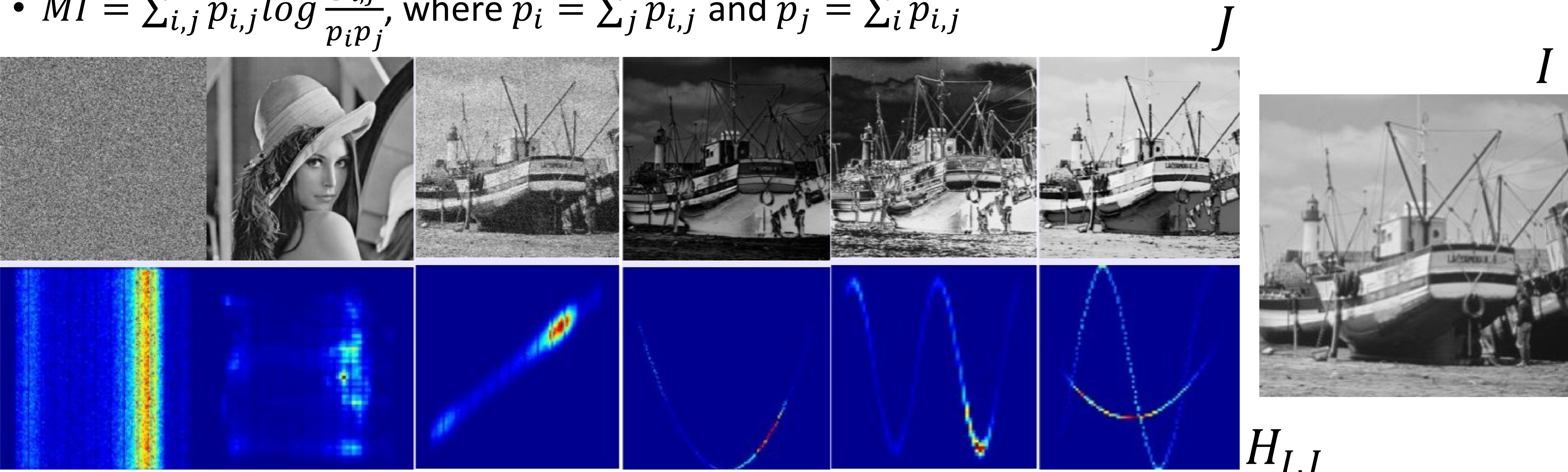


Mutual information

- **similarity criterion based on *statistical* dependence instead of *functional* dependence**
- Mutual information:
- Let X and Y be two random variables. We define *mutual information* as:
- $MI(X,Y) = \sum_{x,y} p(x,y) \log(\frac{p(x,y)}{p(x)p(y)})$
- If X and Y are *statistically independent:* $-p(x,y) = p(x)p(y) \Rightarrow MI(X,Y) = 0$
- $MI(X,Y)$ can be interpreted as 'distance' between $p(x,y)$ and $p(x)p(y)$
    - Also known as Kullback-Leibler *divergence*
- Larger $\mathrm{M}I(X,Y) \Rightarrow X$ and Y share information, or are *statistically dependent*

Mutual information

- normalized joint histogram: $H_{I,J}(i,j) = \frac{1}{n} Card\{(x,y) \mid I(x,y) = i \text{ and } J(x,y) = j\}$
  - $\sum_{i,j} H_{I,J}(i,j) = 1$
  - Now, can interpret $H_{I,J}(i,j)$ as 'probability that given a randomly selected pixel from I with intensity i, the corresponding pixel in image J has intensity j
- can calculate mutual information directly from normalized joint histogram:
- $MI = \sum_{i,j} p_{i,j} log \frac{p_{i,j}}{p_i p_j}$, where $p_i = \sum_j p_{i,j}$ and $p_j = \sum_i p_{i,j}$



$MI = 0.03$ $MI = 0.09$ $MI = 1.22$ $MI = 3.5$ $MI = 1.88$ $MI = 2.7$

Mutual information

- Mutual information decreases as the image is rotated out of alignment with the original
- Top row: rotated images
- Bottom row: joint histogram of rotated image and original image

original