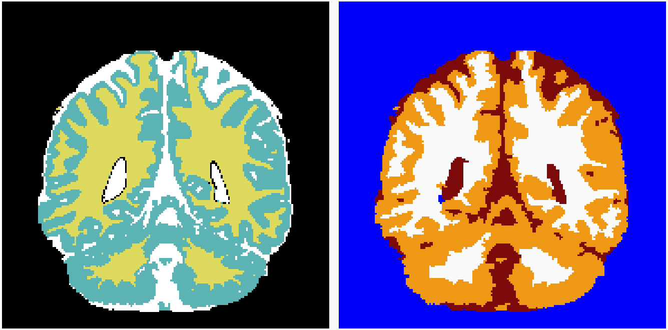
# CatSIM: A Categorical Image Similarity Metric

Geoffrey Thompson and Ranjan Maitra

11/5/2021

# Presenting the Problem

Two images where each pixel is classified into one of several categories. One image (left) is the “ground truth” and the other (right) is a distorted version. How can they be compared?



# Some Ideas (very non-exhaustive)

Binary:

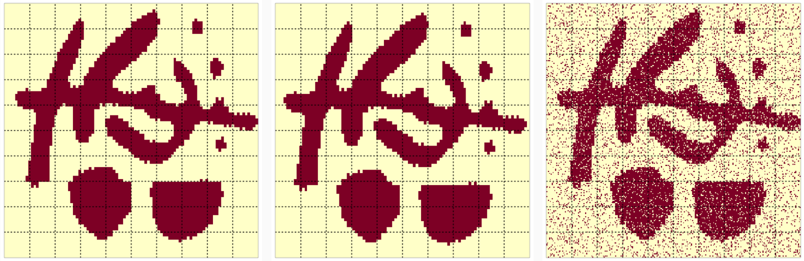
- Accuracy (simple agreement)
- Jaccard Index
- Dice Index

Multi-class:

- Accuracy (simple agreement)
- Cohen's $\kappa$
- Rand or Adjusted Rand Index
- Mutual Information

## What can be improved with these?

These are pixel-by-pixel metrics useful for evaluating clustering solutions. However, the images we are looking at have structure.



The second image is the same as the first except the large blots are shifted down two pixels.

We want a measure that captures this similarity.

There are other measures that try to capture topology, edges, or other non-local similarities.

# Inspiration from Other Image Similarity Metrics

The popular multiscale structural similarity or MS-SSIM for color and grayscale images accounts for spatial and intensity distortions as well as structural information across multiple scales in the image.

The idea is to create an image similarity metric that agrees with human perception of images by looking at structural similarities within the image at multiple scales.

See:

Wang, Z., Simoncelli, E.P., Bovik, A.C. Multiscale Structural Similarity for Image Quality Assessment. In: The Thirty-Seventh Asilomar Conference on Signals, Systems & Computers, 2003, 1398–1402. Pacific Grove, CA, USA: IEEE, 2003. https://doi.org/10.1109/ACSSC.2003.1292216.

# Inspiration (continued)

CW-SSIM is similar, but uses complex wavelets. It is used with binary images, grayscale images, and segmented images. It is designed to deal with image scaling, translation, and rotation.

See:

M. Sampat, Z. Wang, S. Gupta, A. Bovik, and M. Markey, "Complex wavelet structural similarity: A new image similarity index," IEEE Transactions on Image Processing, vol. 18, no. 11, pp. 2385–2401, 11 2009.

# Components of MS-SSIM

When working only on one scale, the SSIM has a "luminance", "contrast", and "structural similarity" function, which compare the local means, local standard deviations, and local covariances of the two images.

$$l(x,y) = \frac{2\mu_x\mu_y + c_1}{\mu_x^2 + \mu_y^2 + c_1}$$

$$c(x,y) = \frac{2\sigma_x\sigma_y + c_2}{\sigma_x^2 + \sigma_y^2 + c_2}$$

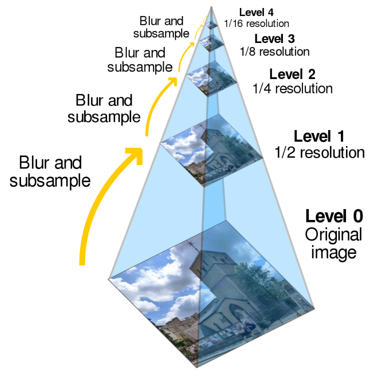
$$s(x,y) = \frac{\sigma_{xy} + c_3}{\sigma_x\sigma_y + c_3}$$

These functions are computed for each $N \times N$ window of pixels in the image and are then combined together for each window:

$$\mathrm{SSIM}(x,y) = \left[l(x,y)^{\alpha} \cdot c(x,y)^{\beta} \cdot s(x,y)^{\gamma}\right]$$

# Image Pyramid

- A type of multiscale representation of images
- Different smoothing options possible - e.g. binomial or Gaussian
- Relevance: how MS-SSIM combines multiple scales of the image



Source: https://commons.wikimedia.org/wiki/File:Image_pyramid.svg by C.M.G. Lee

# MS-SSIM across scales

So far we have defined the MS-SSIM for one scale. To consider structural similarities across scales: downsample by a factor of 2 (with a low-pass filter) and compute the same statistics.

$$\mathrm{MS\text{-}SSIM}(X,Y) = l_M(X,Y)^{\alpha_M} \prod_{j=1}^{M} c_j(X,Y)^{\beta_j} s_j(X,Y)^{\gamma_j}$$

The MS-SSIM as specified by Wang et al uses 5 levels and specific settings of $\alpha$, $\beta$, and $\gamma$ for each level.

The metric aims to capture local variation and structural similarities between images on several scales in a way that mimics the human eye.

# Changing it for the non-continuous context

We need to replace them with statistics for categorical variables rather than continuous ones.

Then we need to combine the measures across different scales.

For a categorical variable, the equivalent of means would be the vector of proportions in each class. We can define a similar categorical variance measure as well.

$$p_i = \frac{1}{n}\{\# x_j = i,\, j \in 1, 2, \ldots, n\} \quad \text{for } i = 1, 2, \ldots, K$$

$$m_x = \{p_i\}_{i=1}^{K}, \quad S_x = \frac{1 - \sqrt{\sum_{i=1}^{K} p_i^2}}{1 - 1/\sqrt{K}} = \frac{1 - \|m_x\|_2}{1 - 1/\sqrt{K}}$$

# Index Measures (continued)

These can then be combined into categorical analogues of the SSIM. We can define, for one level, for each $N \times N$ window of the image:

$$
\begin{aligned}
l^c(x,y) &= \frac{2m_x^\top m_y + C_1}{m_x^\top m_x + m_y^\top m_y + C_1} \\
c^c(x,y) &= \frac{2S_x S_y + C_2}{S_x^2 + S_y^2 + C_2}, \quad s^c(x,y) = v(x,y),
\end{aligned}
$$

where $C_1$, $C_2$, are small scalar constants chosen for numerical stability when the denominator approaches zero, and $v(x,y)$ is an inter-rater agreement measure chosen based on the characteristics of the image.

# Inter-rater agreement

We need an appropriate replacement for the “structural similarity” function which is based on the covariance. This will depend on the nature of the image.

- Jaccard or Dice in binary problems if labels are not symmetrically defined (e.g., activation)
- Accuracy or Cohen’s $\kappa$ if labels are meaningful
- Rand or Adjusted Rand if labels are arbitrarily assigned

# Downsampling

For continuous variables, rescaling is well-understood: downsample and apply a low-pass filter.

Here, we have two options:

1. Take the mode of each $2 \times 2$ block and use a random mode if tied.
2. Instead of downsampling, double the size of the window.

Both give similar results, and the results presented here are option 1.

# Final CatSIM Algorithm

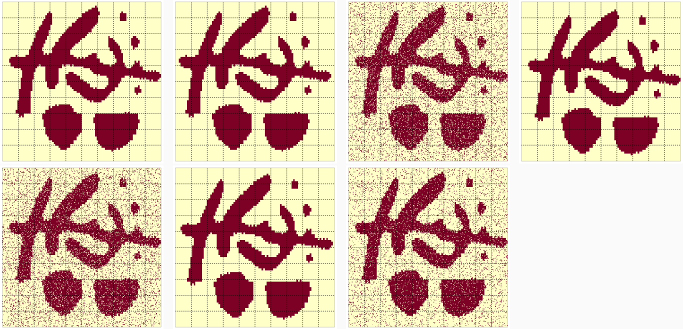
By default, we specify uniform window sizes of $11 \times 11$ and $M = 5$ different levels for 2D applications. We set $\alpha_j = \beta_j = \gamma_j = 1/M$. These parameters can be adjusted based on the application.

1. For two images $X$ and $Y$, the $c^c(x, y)$ and $s^c(x, y)$ statistics are computed over a rolling $N \times N$ pixel (voxel) window and averaged for the entire image while $l^c(x, y)$ is computed for the base level.
2. Downsample each image by a factor of 2.
3. Repeat steps 1 and 2 for each of $M$ total levels.
4. Let $l_1^c(X, Y)$, $c_j^c(X, Y)$ and $s_j^c(X, Y)$ be the average of $l_1^c(X, Y)$, $c_j^c(X, Y)$ and $s_j^c(X, Y)$ over all $N \times N$ windows, for $j = 1, 2, \ldots, M$. Define

$$\mathrm{CatSIM}(X, Y) = [l_1^c(X, Y)]^{\alpha_M} \prod_{j=1}^{M} c_j^c(X, Y)^{\beta_j} s_j^c(X, Y)^{\gamma_j},$$

# Illustration: Besag (1986) Binary Image

The original 88 × 100 image has been blown up to a 264 × 300 image. We added spatial translations and salt-and-pepper noise to match the error in the spatially-translated images. We compute the default CatSIM, CatSIM with only one level, CatSIM with a window equal to the entire image, and the Adjusted Rand ($AR$), Cohen's $\kappa$, and CW-SSIM.



# Illustration: Besag (1986) Binary Image

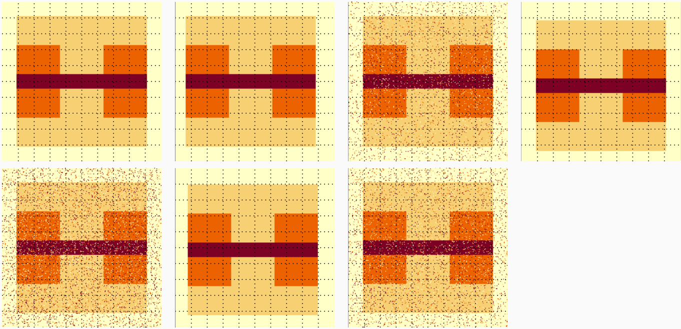
**Table 1:** CatSIM ($\kappa$) and other metrics for different distortions.

| Image | CatSIM 5 levels | CatSIM 1 level | CatSIM (whole) | $AR$ | $\kappa$ | CW-SSIM |
|---|---|---|---|---|---|---|
| H Shift | 0.594 | 0.464 | 0.763 | 0.627 | 0.763 | 0.831 |
| H - S & P | 0.515 | 0.092 | 0.769 | 0.630 | 0.771 | 0.783 |
| V Shift | 0.569 | 0.449 | 0.751 | 0.610 | 0.751 | 0.752 |
| V - S & P | 0.516 | 0.090 | 0.756 | 0.613 | 0.759 | 0.780 |
| H & V Shift | 0.658 | 0.561 | 0.827 | 0.720 | 0.827 | 0.834 |
| H & V - S & P | 0.557 | 0.110 | 0.832 | 0.725 | 0.834 | 0.810 |

CatSIM rates the spatially-shifted images differently from the S&P-degraded images.

# Illustration: Constructed Image

Here we construct a 220 × 220 image with 4 classes. We added spatial translations and salt-and-pepper noise to match the error in the spatially-translated images. We compute for comparison the default CatSIM, CatSIM with only one level, CatSIM with a window equal to the entire image, and the Adjusted Rand ($AR$) and Cohen's $\kappa$. The true image is on the left.



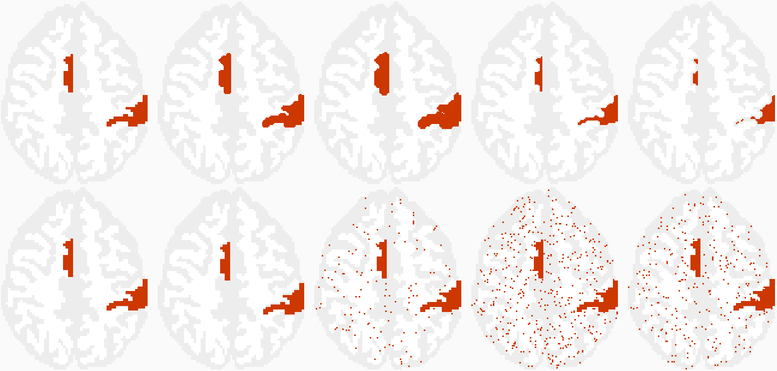
# Illustration: Constructed Image Results

| Image | CatSIM 5 levels | CatSIM 1 level | CatSIM (whole) | $AR$ | $\kappa$ |
|---|---|---|---|---|---|
| H Shift | 0.816 | 0.686 | 0.906 | 0.828 | 0.906 |
| H - S & P | 0.610 | 0.105 | 0.906 | 0.842 | 0.907 |
| V Shift | 0.652 | 0.462 | 0.827 | 0.723 | 0.827 |
| V - S & P | 0.548 | 0.079 | 0.825 | 0.717 | 0.827 |
| H & V Shift | 0.814 | 0.533 | 0.869 | 0.777 | 0.869 |
| H & V - S & P | 0.565 | 0.093 | 0.874 | 0.790 | 0.875 |

The default 5-level and 1-level CatSIM indices clearly distinguish between images that are minor spatial perturbations over images degraded with added noise.

# Illustration: Highly-Imbalanced Binary Image

Our ground truth is a 256×256 version of the modified 128×128 Hoffman activation phantom that has a small proportion (3.98%) of truly activated in-brain pixels. We dilate, erode, shift, and add salt-and-pepper noise.



# Illustration: Highly-Imbalanced Binary Image

|  | $J$ | Dice | Acc. | CatSIM ($J$) | CatSIM ($\kappa$) |
|---|---|---|---|---|---|
| Dilated (D+1) | 0.75 | 0.86 | 0.99 | 0.74 | 0.94 |
| Dilated (D+2) | 0.61 | 0.76 | 0.97 | 0.59 | 0.88 |
| Eroded (E-1) | 0.68 | 0.81 | 0.99 | 0.61 | 0.90 |
| Eroded (E-2) | 0.43 | 0.60 | 0.98 | 0.33 | 0.72 |
| Shift Up (S↑1) | 0.83 | 0.91 | 0.99 | 0.75 | 0.94 |
| Shift Down (S↓2) | 0.68 | 0.81 | 0.99 | 0.52 | 0.77 |
| Noise ($+\frac{\varepsilon}{100}$) | 0.81 | 0.90 | 0.99 | 0.63 | 0.80 |
| Noise ($+\frac{3\varepsilon}{100}$) | 0.57 | 0.73 | 0.97 | 0.45 | 0.58 |
| Shift+Noise (S↑1$+\frac{2\varepsilon}{100}$) | 0.58 | 0.73 | 0.97 | 0.42 | 0.63 |

Compared to the Jaccard index, CatSIM($J$) penalizes noise in the images more than for minor perturbations that do not affect the basic spatial extent of the activated region.

# Illustration: Highly-Imbalanced Binary Image

Here we investigate the utility of using 5 layers.

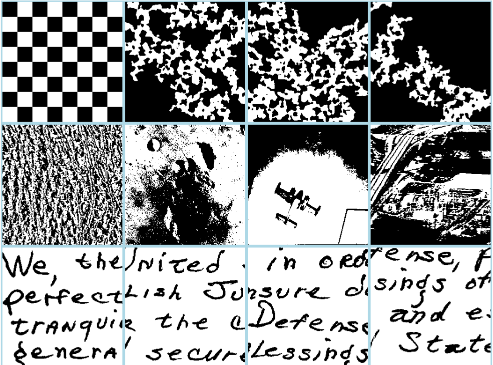
**Table 2:** CatSIM ($J$) values for each layer for different types of distortions. Layer 1 is the image itself subsequent layers downsample by a factor of 2.

| | Layer 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| Dilated (D+1) | 0.60 | 0.65 | 0.68 | 0.83 | 1.00 |
| Dilated (D+2) | 0.43 | 0.48 | 0.53 | 0.62 | 1.00 |
| Eroded (E-1) | 0.52 | 0.51 | 0.57 | 0.56 | 1.00 |
| Eroded (E-2) | 0.28 | 0.32 | 0.34 | 0.38 | 0.35 |
| Shift Up (S↑1) | 0.72 | 0.59 | 0.72 | 0.75 | 1.00 |
| Shift Down (S↓2) | 0.55 | 0.59 | 0.58 | 0.59 | 0.35 |
| Noise ($+\frac{\varepsilon}{100}$) | 0.12 | 0.86 | 1.00 | 1.00 | 1.00 |
| Noise ($+\frac{3\varepsilon}{100}$) | 0.04 | 0.46 | 1.00 | 1.00 | 1.00 |
| Shift+Noise ($S\uparrow 1+\frac{2\varepsilon}{100}$) | 0.05 | 0.48 | 0.79 | 0.75 | 1.00 |

The first level rates added noise poorly, but higher levels smooth out that difference. Large differences, like double erosion (E-2), damage the image's rating across all scales. Any larger scale than this has ratings of either 1 or 0 as the activated class disappears completely.

# Image Quality Assessment Surveys

For the twelve binary images shown here, 12 distorted versions of each were created and 74 adult volunteers were randomly shown 30 distorted images and their corresponding undistorted version and asked to rate the quality of each distorted image on a scale from 1 to 100, with 100. We then calculate a mean opinion score (MOS) for each image.



# Image Quality Assessment Surveys: Binary Results

We compared the MOS to CW-SSIM, the space-unaware metrics of $\kappa$, $AR$, Jaccard, and accuracy (that can be related to Peak-Signal-to-Noise-Ratio) and the CatSIM metrics with $\kappa$, $AR$, $J$ and accuracy.

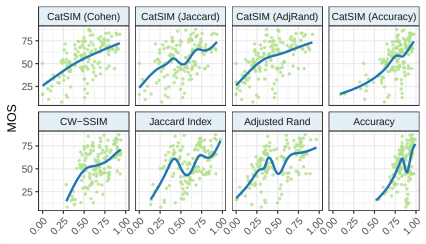


**Table 3:** Correlation with MOS for each metric.

| Method | $\rho$ |
|---|---|
| CatSIM (Acc) | 0.601 |
| CatSIM ($\kappa$) | 0.598 |
| CatSIM ($AR$) | 0.580 |
| MS-SSIM | 0.578 |
| Cohen's $\kappa$ | 0.577 |
| $AR$ | 0.557 |
| CW-SSIM | 0.510 |
| Accuracy | 0.500 |
| CatSIM ($J$) | 0.470 |
| Jaccard | 0.464 |

# Image Quality Assessment Surveys: Binary Results

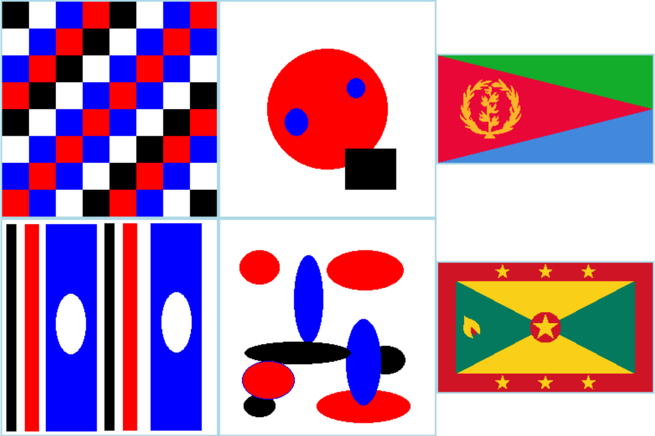
**Table 4:** P-values for the randomization test of whether the CatSIM methods are more correlated with MOS.

| | CatSIM ($\kappa$) | CatSIM ($AR$) | CatSIM (Acc) |
|---|---|---|---|
| MS-SSIM | 0.311 | 0.424 | 0.410 |
| CW-SSIM | **0.018** | 0.082 | 0.069 |
| Accuracy | **0.032** | **0.029** | **0.033** |
| Cohen's $\kappa$ | 0.203 | 0.369 | 0.348 |
| $AR$ | 0.123 | 0.246 | 0.204 |

These metrics are all positively correlated with the MOS, with CatSIM methods using $\kappa$, $AR$ and accuracy performing the best. A randomization test indicated significantly higher correlations with the MOS for CatSIM ($\kappa$) against CW-SSIM and accuracy and for both CatSIM ($AR$) and CatSIM (Accuracy) against accuracy.

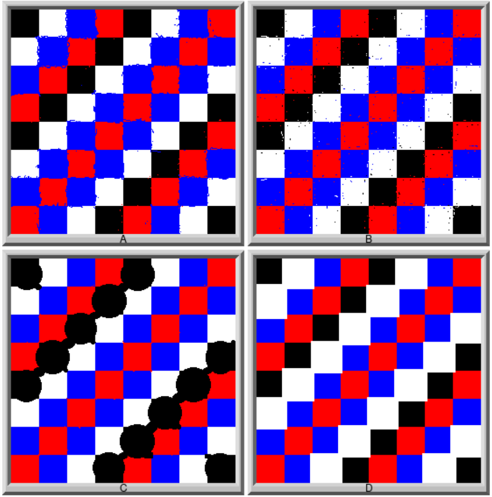
# Image Quality Assessment Surveys: Categorical Images

Each of 614 adult volunteers were shown 11 sets of 4 distorted images along with the ground truth and asked to rank the distorted images in each set from most to least similar to the original.



# Image Quality Assessment Surveys: Categorical Images

An example of a panel of four distorted images shown to volunteers:



# Image Quality Assessment Surveys: Categorical Images

The table reports the squared difference between the mean rankings of the sets of images by the raters and by the different metrics (the CatSIM variants and the space-unaware accuracy, Rand, *AR* and $\kappa$).

**Table 5:** Squared differences in mean rankings produced by human raters in the survey and the rankings produced by similarity metrics.

| Method | Squared Difference | RMSE |
|---|---|---|
| CatSIM (Accuracy) | 72.385 | 0.1933 |
| CatSIM (Rand) | 74.434 | 0.1961 |
| CatSIM ($\kappa$) | 77.695 | 0.2003 |
| MOS | 79.848 | 0.2030 |
| Adjusted Rand Index | 81.525 | 0.2052 |
| Cohen's $\kappa$ | 81.548 | 0.2052 |
| Rand Index | 85.014 | 0.2096 |
| Accuracy | 85.461 | 0.2101 |
| CatSIM (Adj Rand) | 87.838 | 0.2130 |

In this experiment, the CatSIM methods using accuracy, the Rand index, or Cohen's $\kappa$ produce rankings more similar to those produced by human raters

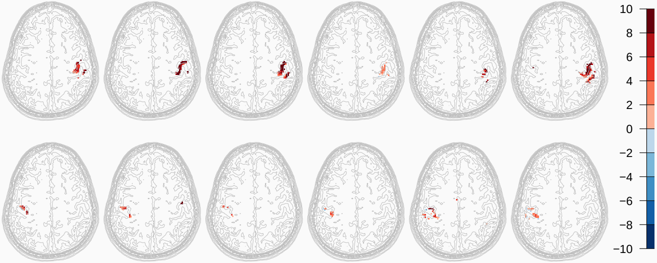
# Applications to Real Data

- Assessing Test-Retest Reliability of Activation in fMRI
- Evaluating Image Segmentation Algorithms

# Test-Retest Reliability of Activation in fMRI

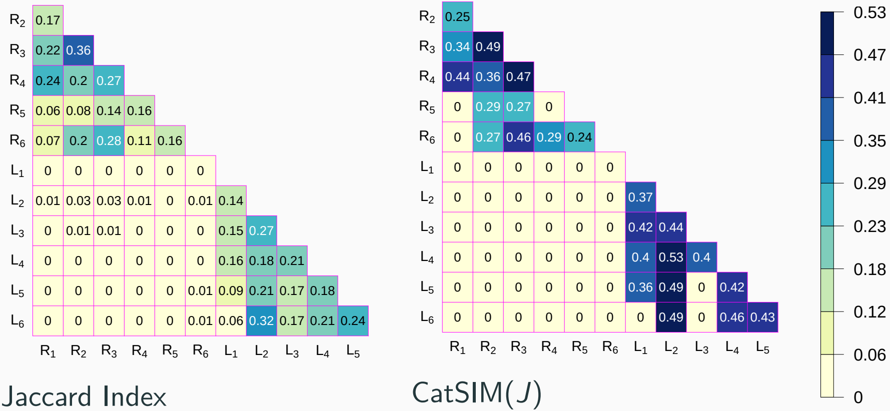
Repeatability of results across multiple fMRI studies is important.

Our data are from the replicated right- and left-hand finger-tapping experiments in which activation was detected using the AR-FAST algorithm.



Activation images of the 20th slice in the finger-tapping experiments. The top row is right-hand and the bottom is left-hand experiments.

# Test-Retest Reliability Continued



|  | $R_1$ | $R_2$ | $R_3$ | $R_4$ | $R_5$ | $R_6$ | $L_1$ | $L_2$ | $L_3$ | $L_4$ | $L_5$ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $R_2$ | 0.17 |  |  |  |  |  |  |  |  |  |  |
| $R_3$ | 0.22 | 0.36 |  |  |  |  |  |  |  |  |  |
| $R_4$ | 0.24 | 0.2 | 0.27 |  |  |  |  |  |  |  |  |
| $R_5$ | 0.06 | 0.08 | 0.14 | 0.16 |  |  |  |  |  |  |  |
| $R_6$ | 0.07 | 0.2 | 0.28 | 0.11 | 0.16 |  |  |  |  |  |  |
| $L_1$ | 0 | 0 | 0 | 0 | 0 | 0 |  |  |  |  |  |
| $L_2$ | 0.01 | 0.03 | 0.03 | 0.01 | 0 | 0.01 | 0.14 |  |  |  |  |
| $L_3$ | 0 | 0.01 | 0.01 | 0 | 0 | 0 | 0.15 | 0.27 |  |  |  |
| $L_4$ | 0 | 0 | 0 | 0 | 0 | 0 | 0.16 | 0.18 | 0.21 |  |  |
| $L_5$ | 0 | 0 | 0 | 0 | 0 | 0.01 | 0.09 | 0.21 | 0.17 | 0.18 |  |
| $L_6$ | 0 | 0 | 0 | 0 | 0 | 0.01 | 0.06 | 0.32 | 0.17 | 0.21 | 0.24 |

Jaccard Index

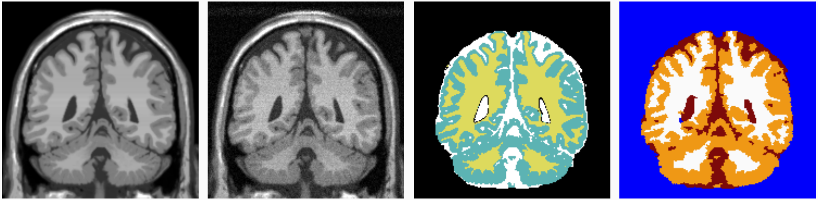
|  | $R_1$ | $R_2$ | $R_3$ | $R_4$ | $R_5$ | $R_6$ | $L_1$ | $L_2$ | $L_3$ | $L_4$ | $L_5$ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $R_2$ | 0.25 |  |  |  |  |  |  |  |  |  |  |
| $R_3$ | 0.34 | 0.49 |  |  |  |  |  |  |  |  |  |
| $R_4$ | 0.44 | 0.36 | 0.47 |  |  |  |  |  |  |  |  |
| $R_5$ | 0 | 0.29 | 0.27 | 0 |  |  |  |  |  |  |  |
| $R_6$ | 0 | 0.27 | 0.46 | 0.29 | 0.24 |  |  |  |  |  |  |
| $L_1$ | 0 | 0 | 0 | 0 | 0 | 0 |  |  |  |  |  |
| $L_2$ | 0 | 0 | 0 | 0 | 0 | 0 | 0.37 |  |  |  |  |
| $L_3$ | 0 | 0 | 0 | 0 | 0 | 0 | 0.42 | 0.44 |  |  |  |
| $L_4$ | 0 | 0 | 0 | 0 | 0 | 0 | 0.4 | 0.53 | 0.4 |  |  |
| $L_5$ | 0 | 0 | 0 | 0 | 0 | 0 | 0.36 | 0.49 | 0 | 0.42 |  |
| $L_6$ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.49 | 0 | 0.46 | 0.43 |

CatSIM($J$)

Graphical displays of $J$ and CatSIM($J$) values for each 3D volume pair, with $R_i$ or $L_i$ indicating $i$th right- or left-hand experiment.

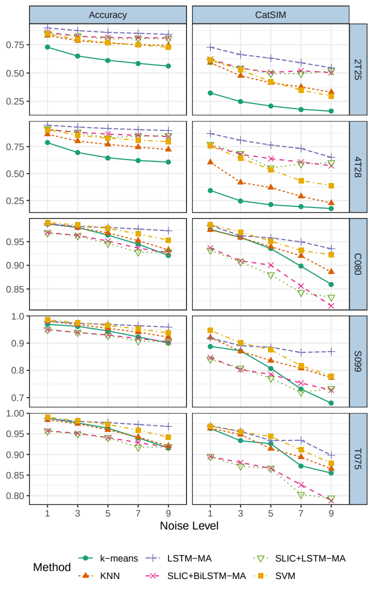
# Evaluating Image Segmentation

Segmenting Magnetic Resonance (MR) images into regions of gray matter, white matter, or cerebrospinal fluid is important for diagnostic purposes and important for automated image processing. We demonstrate CatSIM on a practical example using data made available by Xie and Wen (2019) who evaluate their new segmentation algorithm using simulated datasets from BrainWeb, and real-data images from MRBrainS.



Baseline image C080 from the BrainWeb data set, image C080 with 5% Rayleigh noise added, Ground Truth segmentation of image C080, predicted SLIC-BiLSTM-MA segmentation.

# Evaluating Image Segmentation



Xie and Wen provide five examples, three from the BrainWeb data set, and two from the MRBrainS data set. We see that the CatSIM metric gives, for almost all noise settings and across all images, the same ordering of the methods as the accuracy. However, the spread of the results is greater, meaning that, as before, we get better discrimination between the methods using CatSIM than with pointwise accuracy.

# Conclusion

- Novel method for comparing binary and multinary images in two and three dimensions
- Provides results more similar to human perception in ranking images
- Provides greater discrimination between segmentations than currently used metrics.

Future work:

- Smooth windowing functions
- Fuzzy class labels
- Different misclassification costs
- Hierarchical class information