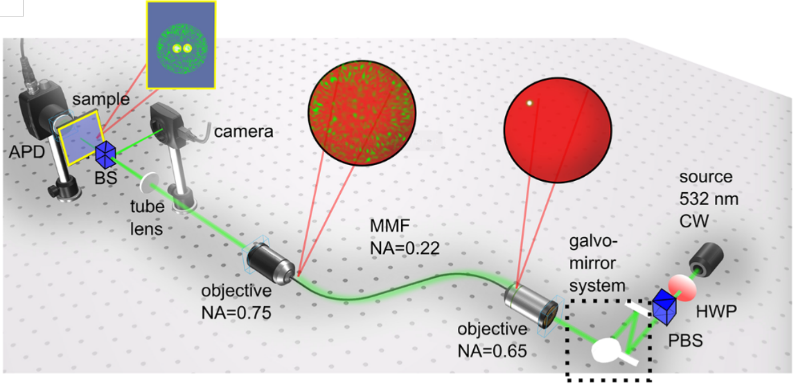
Sparsity-inducing bases for compressive sensing

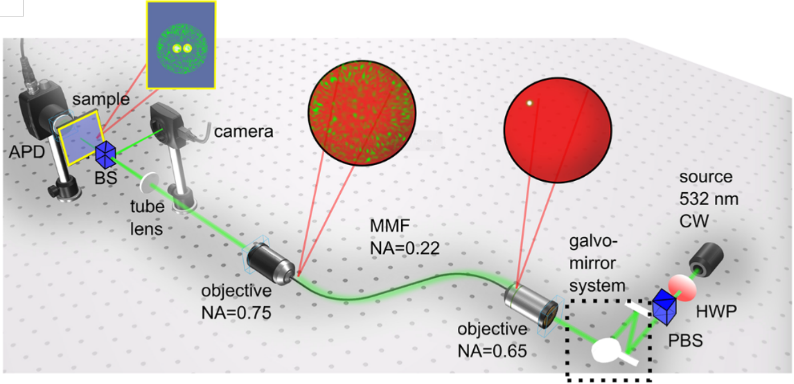
Nathan van den Berg Martijn Brouwer Robin van der Laag
Marcelo Geurts Galdámez Sean Tobin

Supervised by:
Ksenia Abrashitova (ARCNL), Tristan van Leeuwen (UU),
Rob Bisseling (UU), Ivan Kryven (UU)

21 April 2023

Experimental set-up





$$y = \sum_{j=1}^{n^2} a_j \cdot x_j = \langle a, x \rangle \qquad y_i = \langle a_i, x \rangle$$

Mathematical formulation

$$Ax = \begin{bmatrix} \dots & a_1 & \dots \\ & \vdots & \\ \dots & a_m & \dots \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_{n^2} \end{bmatrix} = \begin{bmatrix} y_1 \\ \vdots \\ y_m \end{bmatrix} = y, \text{ where } A \in \mathbb{R}^{m \times n^2}.$$

Mathematical formulation

$$Ax = \begin{bmatrix} \dots & a_1 & \dots \\ & \vdots & \\ \dots & a_m & \dots \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_{n^2} \end{bmatrix} = \begin{bmatrix} y_1 \\ \vdots \\ y_m \end{bmatrix} = y, \text{ where } A \in \mathbb{R}^{m \times n^2}.$$

$$\underset{x \in \mathbb{R}^{n^2}}{\arg\min} \; \|Ax - y\|_2^2 + \lambda \|x\|_1 \qquad x = \mathrm{GPSR}(A, y, \lambda)$$

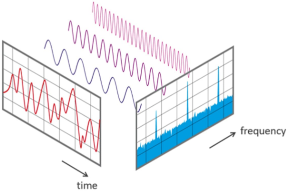
$$Ax = \begin{bmatrix} \dots & a_1 & \dots \\ & \vdots & \\ \dots & a_m & \dots \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_{n^2} \end{bmatrix} = \begin{bmatrix} y_1 \\ \vdots \\ y_m \end{bmatrix} = y, \text{ where } A \in \mathbb{R}^{m \times n^2}.$$

$$\underset{x \in \mathbb{R}^{n^2}}{\arg\min} \|Ax - y\|_2^2 + \lambda \|x\|_1 \qquad x = \mathrm{GPSR}(A, y, \lambda)$$

$$\underset{x \in \mathbb{R}^{n^2}}{\arg\min} \|(AB^{-1})(Bx) - y\|_2^2 + \lambda \|Bx\|_1 \qquad Bx = \mathrm{GPSR}(AB^{-1}, y, \lambda)$$

Goal: GPSR performs better when x is sparse, so we try to find B such that Bx is sparse.

Discrete Cosine Transform



$$x_k^c = \sum_{n=0}^{M-1} x_n \cos\left(\frac{\pi}{M}\left(n+\frac{1}{2}\right)k\right)$$

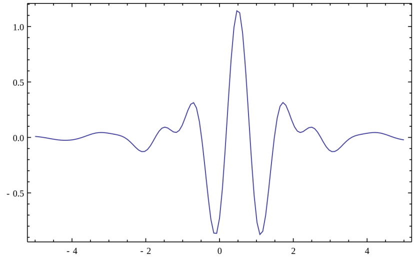
$$x^c = \begin{bmatrix} 1 & 1 & \dots & 1 \\ \cos\left(\frac{\pi}{2M}\right) & \cos\left(\frac{3\pi}{2M}\right) & \dots & \cos\left(\frac{(M-\frac{1}{2})\pi}{M}\right) \\ \vdots & \vdots & \ddots & \vdots \\ \cos\left(\frac{(M-1)\pi}{2M}\right) & \cos\left(\frac{(M-1)3\pi}{2M}\right) & \dots & \cos\left(\frac{(M-1)(M-\frac{1}{2})\pi}{M}\right) \end{bmatrix} x$$

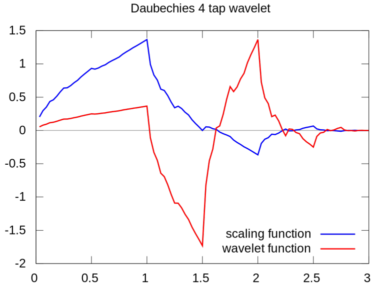
Discrete Cosine Transform

$$X^c = C(CX^T)^T$$

$$\begin{array}{cc} \text{Row 1} & \\ \text{Row 2} & \\ \text{Row 3} & \end{array} \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \qquad \begin{array}{c} \begin{array}{ccc} \text{Column 1} & \text{Column 1} & \text{Column 1} \end{array} \\ \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \end{array}$$

DAUB4 Wavelet Transform





$$
\begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \end{bmatrix}
\xrightarrow{W_{\text{coef}}(8)}
\begin{bmatrix} x'_0 \\ x'_1 \\ x'_2 \\ x'_3 \\ x'_4 \\ x'_5 \\ x'_6 \\ x'_7 \end{bmatrix}
\xrightarrow{P(8)}
\left[\begin{array}{c} x'_0 \\ x'_2 \\ x'_4 \\ x'_6 \\ \hline x'_1 \\ x'_3 \\ x'_5 \\ x'_7 \end{array}\right]
\xrightarrow{W_{\text{coef}}(4)}
\left[\begin{array}{c} x''_0 \\ x''_2 \\ x''_4 \\ x''_6 \\ \hline x'_1 \\ x'_3 \\ x'_5 \\ x'_7 \end{array}\right]
\xrightarrow{P(4)}
\left[\begin{array}{c} x''_0 \\ x''_4 \\ \hline x''_2 \\ x''_6 \\ \hline x'_1 \\ x'_3 \\ x'_5 \\ x'_7 \end{array}\right]
\xrightarrow{W_{\text{coef}}(2)}
\left[\begin{array}{c} x'''_0 \\ x'''_4 \\ \hline x''_2 \\ x''_6 \\ \hline x'_1 \\ x'_3 \\ x'_5 \\ x'_7 \end{array}\right]
$$

We want to find an $\hat{x} \in \mathbb{R}^{N^2}$, with $\hat{x} = Bx$ that is a solution to

$$\min_{\hat{x} \in \mathbb{R}^{n^2}} \|AB^{-1}\hat{x} - y\|_2^2 + \lambda \|\hat{x}\|.$$

Traditional Transforms

We want to find an $\hat{x} \in \mathbb{R}^{N^2}$, with $\hat{x} = Bx$ that is a solution to

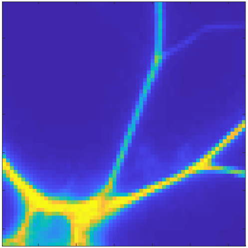
$$\min_{\hat{x} \in \mathbb{R}^{n^2}} \|AB^{-1}\hat{x} - y\|_2^2 + \lambda \|\hat{x}\|.$$

Using four different transforms B^{-1}:

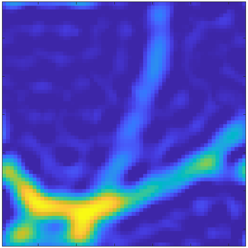
1. $B^{-1} = I$, no transformation,
2. $B^{-1} = C^{-1} \otimes C^{-1}$, with C the discrete cosine transformation matrix,
3. $B^{-1} = W^{-1} \otimes W^{-1}$, with W the Daubechies-4 wavelet transformation matrix,
4. $B^{-1} = (C \otimes C)(W^{-1} \otimes W^{-1})$, with C and W the same as above.

Recovered Images

Sample

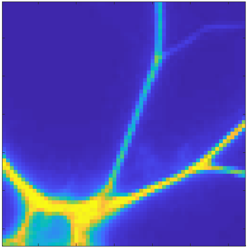


Baseline

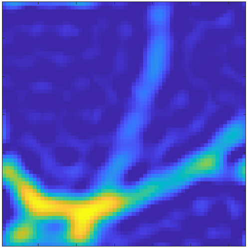


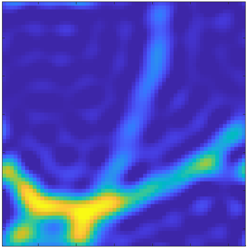
Recovered Images

Sample

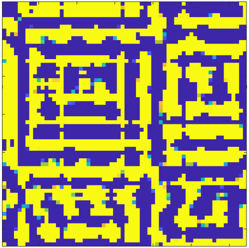


Baseline

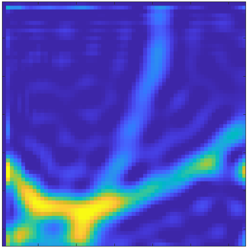




DCT

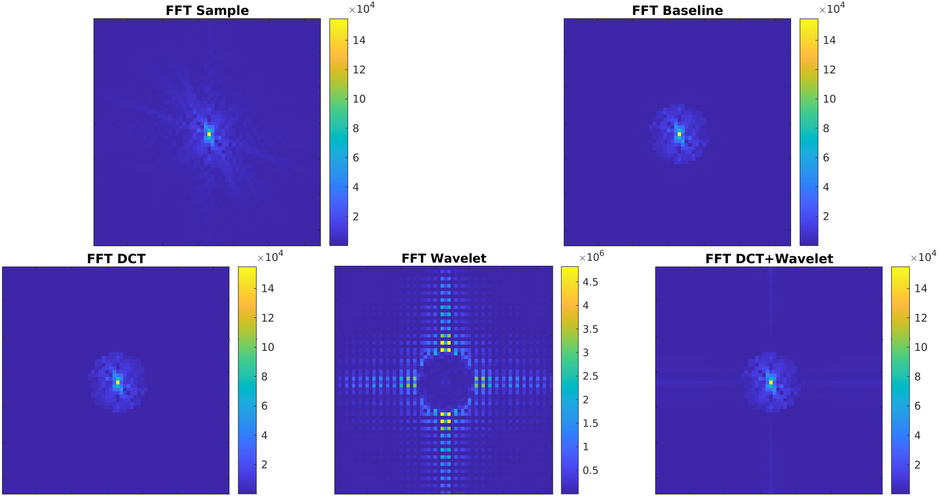


Wavelet



DCT + Wavelet

Fourier Transformed Images



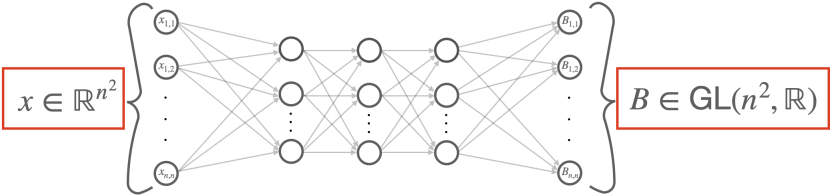
Goal: find a basis transformation B such that the sparsity of Bx increases and with which the reconstruction quality using GPSR.

Machine learning approach

Goal: find a basis transformation B such that the sparsity of Bx increases and with which the reconstruction quality using GPSR.

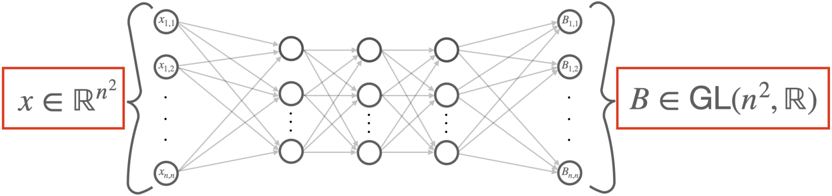
Machine learning approach: train a neural network with sparsity-inducing loss function that preserves orthogonality with flattened training samples x as input.

The network



$x \in \mathbb{R}^{n^2}$
$x_{1,1}$
$x_{1,2}$
$x_{n,n}$
$B_{1,1}$
$B_{1,2}$
$B_{n,n}$
$B \in \mathrm{GL}(n^2, \mathbb{R})$

The network



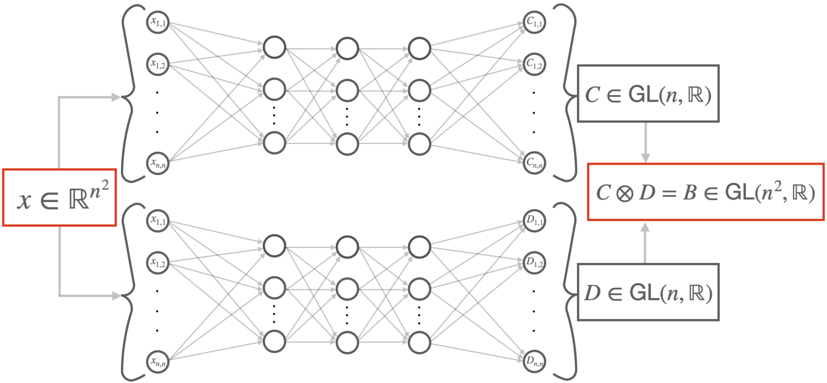
1. $loss_{sparse}[Bx] = \sum_i \sqrt{(Bx)_i^2 + \epsilon} \approx \|Bx\|_1, \quad \epsilon > 0$ (small).
2. $loss_{orth.}[B] = \|BB^\top - I_n\|_2^2 = \mathrm{MSE}(BB^\top - I_n)$.

Dimension reduction

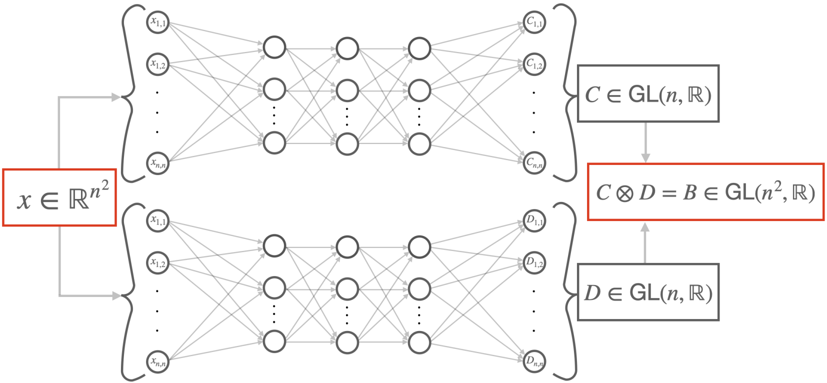
Let $C = [c_{ij}]_{i,j} \in \mathbb{R}^{n \times n}$ and $D = [d_{ij}]_{i,j} \in \mathbb{R}^{n \times n}$.
The Kronecker product of C and D is defined as,

$$B := C \otimes D = \begin{bmatrix} c_{11}D & c_{12}D & \dots & c_{1n}D \\ c_{21}D & c_{22}D & \dots & c_{2n}D \\ \vdots & \vdots & \ddots & \vdots \\ c_{n1}D & c_{n2}D & \dots & c_{nn}D \end{bmatrix} \in \mathbb{R}^{n^2 \times n^2}.$$

Two linked neural networks



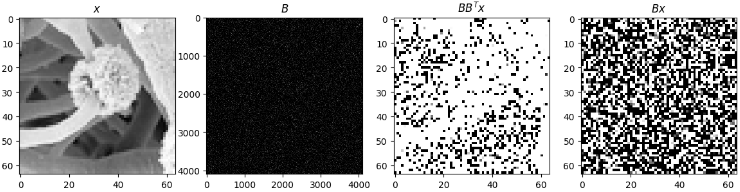
Two linked neural networks



$$loss(x; C, D) = loss_{sparse}[(C \otimes D)x] + \alpha\left(loss_{orth.}[C] + loss_{orth.}[D]\right)$$

$$= \sum_i \sqrt{[(C \otimes D)x]_i^2 + \epsilon} + \alpha\left(\|CC^\top - I_n\|_2^2 + \|DD^\top - I_n\|_2^2\right).$$

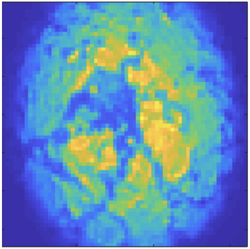
Training

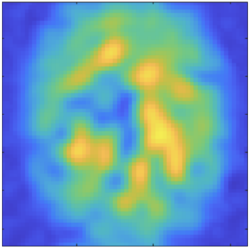


Iteration: 100
Sparsity x: 1
Sparsity Bx: 1993

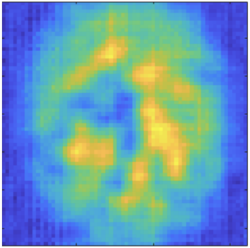
Results

Sample

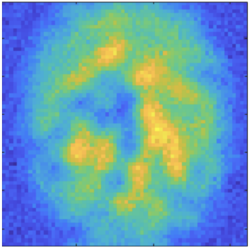




Identity B



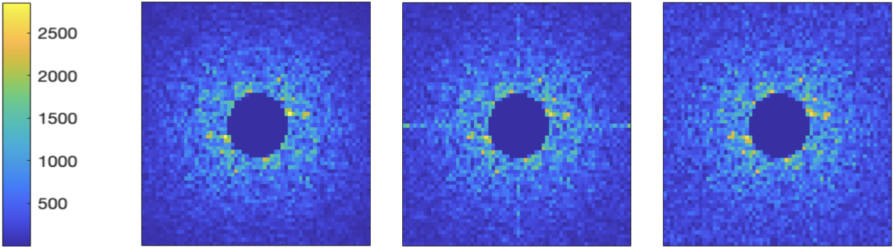
Kronecker B



Regular B

Quantifying errors

	Identity B	Kronecker B	Regular B
Correlation	0.93939	0.93393	0.9267
S.similarity	0.52892	0.47649	0.44159
FFT error	1362159.6975	1462938.2558	1640305.0802



Conclusions

1. The GPSR algorithm fails to reproduce the high-frequency behaviour of the original sample.

Conclusions

1. The GPSR algorithm fails to reproduce the high-frequency behaviour of the original sample.
2. The DCT and wavelet transforms seem to not increase the accuracy of the reconstructed images enough to warrant their use.

Conclusions

1. The GPSR algorithm fails to reproduce the high-frequency behaviour of the original sample.
2. The DCT and wavelet transforms seem to not increase the accuracy of the reconstructed images enough to warrant their use.
3. Training sparsity-inducing networks is time consuming, but yields promising results provided that more improved algorithms are developed.

Conclusions

1. The GPSR algorithm fails to reproduce the high-frequency behaviour of the original sample.
2. The DCT and wavelet transforms seem to not increase the accuracy of the reconstructed images enough to warrant their use.
3. Training sparsity-inducing networks is time consuming, but yields promising results provided that more improved algorithms are developed.

Questions?

2*Astrocyte	GPSR Output		Recovered Image	
	corr2	ssim	corr2	ssim
Baseline	NA	NA	0.91749	0.25872
DCT	0.94775	0.00716	0.92562	0.27034
DAUB4	-0.03108	-8.2600-05	-0.02966	-1.3560-05
DAUB4 + DCT	0.94489	0.23078	0.90678	0.23078

Table: 2D Correlation and the Structural similarity of the recovered sample/image using the different transformations on the 64×64 Astrocyte sample.

BONUS SLIDE2 (CIFAR10)