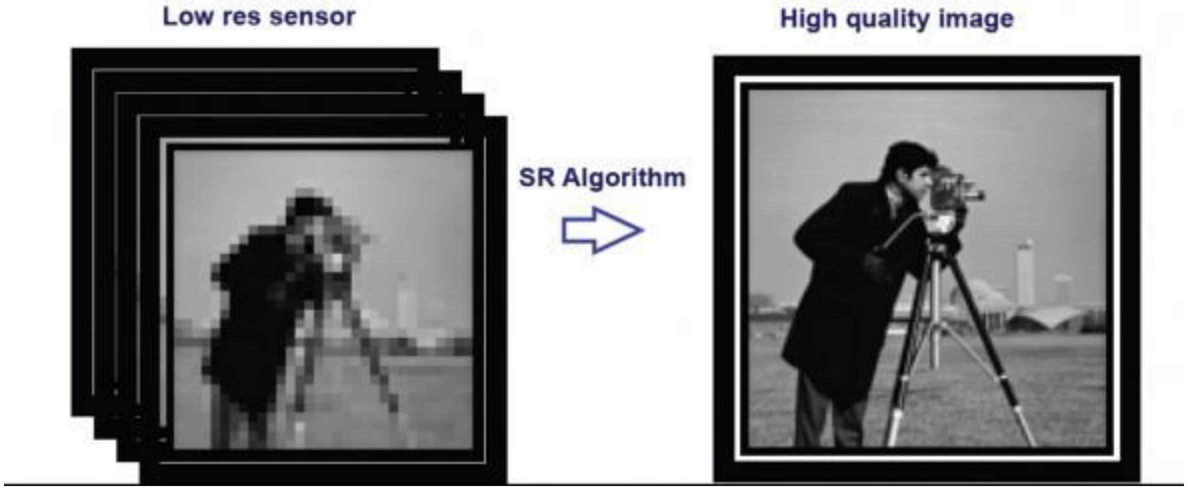
MRI Super Resolution Problem

Rinu Sebastian

Reference: Shi, Feng, et al. "LRTV: MR image super-resolution with low-rank and total variation regularizations." *IEEE TMI* (2015)

What is Super Resolution Problem?



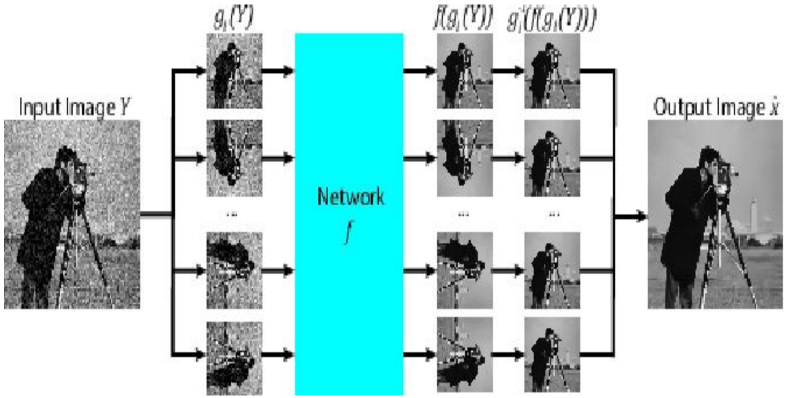
https://amundtveit.com/2017/06/04/deep-learning-for-image-super-resolution-scale-up/

Super Resolution

Scope:

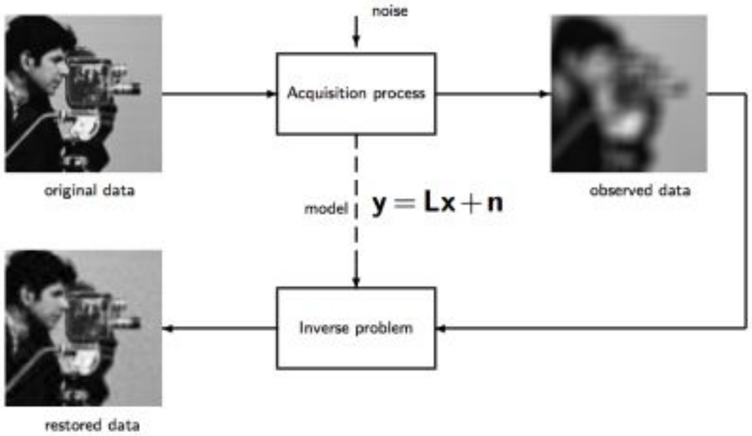
Approach:

CNN



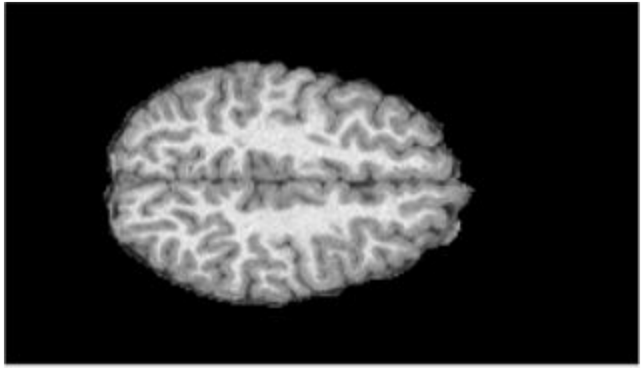
https://www.semanticscholar.org/paper/Self-Commmittee-Approach-for-Image-Restoration-Ahn-Cho/7c7db96ac61642a49da556428a6c7c16e34bb3c4

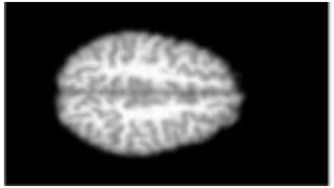
Inverse Problem

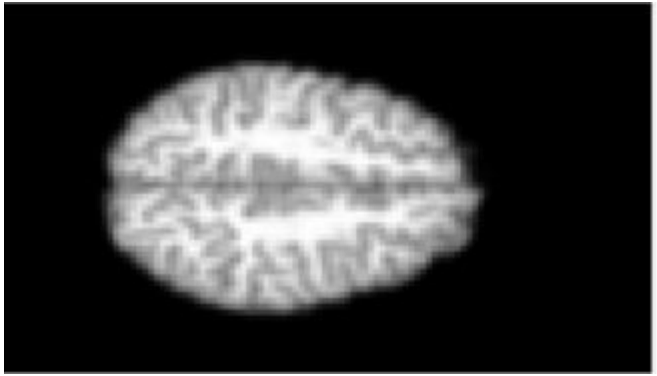


http://bigwww.epfl.ch/teaching/projects/abstracts/guerquin/master.html

Inverse Problem







$$T = DSX + n$$

down-sampling

blurring

Observation noise

Reference: Shi, Feng, et al. "LRTV: MR image super-resolution with low-rank and total variation regularizations." *IEEE TMI* (2015).

LRTV : Low Rank Total-Variation

for a single image SR:

$$\widehat{X} = \arg\min_{X} \|DSX - T\|^2.$$

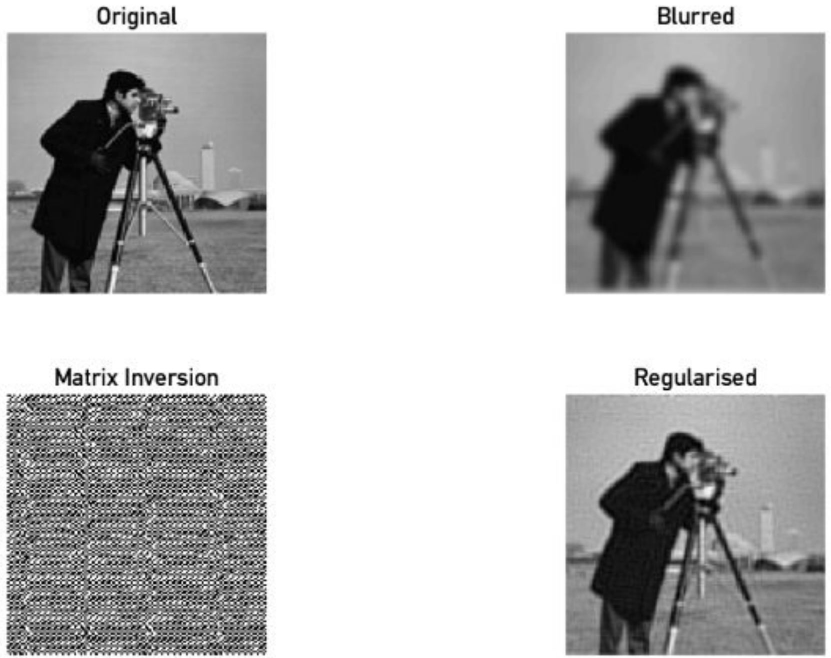
Few issues:

1. Ill-posed inverse problem
2. Not using prior knowledge
3. Edge preservation
4. Utilises information around the near-by voxels.

Solution:

$$\widehat{X} = \arg\min_{X} \|DSX - T\|^2 + \lambda_{rank} Rank(X) + \lambda_{tv} TV(X)$$

Total Variation Regularisation

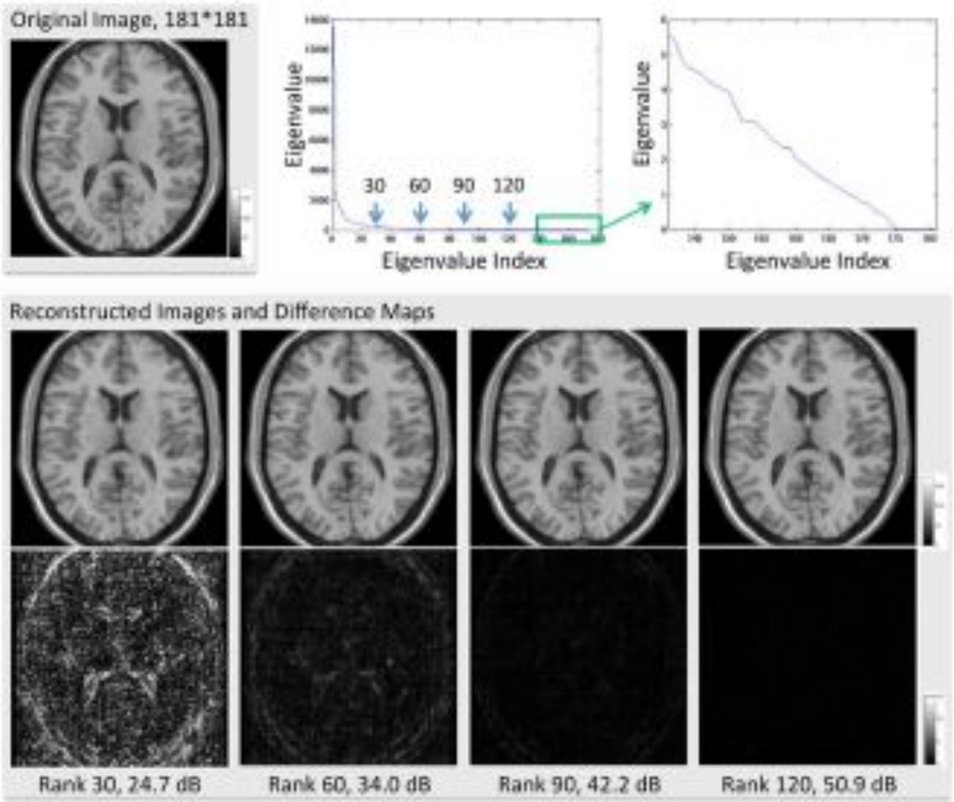


https://jasmcole.com/2016/06/12/regularisation/

Eigenvalue decomposition

$$A \quad = P\Sigma Q^T = (\mathbf{p_1}, \mathbf{p_2}, \cdots, \mathbf{p_n}) \begin{pmatrix} \sigma_1 & 0 & \cdots & 0 \\ 0 & \sigma_2 & \ddots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & \sigma_n \end{pmatrix} \begin{pmatrix} \mathbf{q_1}^T \\ \mathbf{q_2}^T \\ \vdots \\ \mathbf{q_n}^T \end{pmatrix}$$

Low Rank Completion



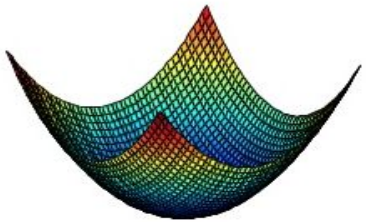
Reference: Shi, Feng, et al. "LRTV: MR image super-resolution with low-rank and total variation regularizations." *IEEE TMI* (2015)

Convex Optimization

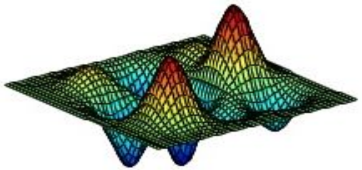
Disclaimer: Math ahead! :)
Optimization Problem:

$$\hat{X} = \arg\min_{X} \|DSX - T\|^2 + \lambda_{rank} Rank(X) + \lambda_{tv} TV(X)$$

Convex optimization



Non-convex optimization



$$Rank(X) = \sum_{i=1}^{N} \alpha_i \|X_{(i)}\|_{tr}$$

X= U*V*W => X(1)= U*(V*W) X(2) = V*(W*U) X(3)=W*(U*V)

Convex Optimization Problem:

$$\min_{X, \{M_i\}_{i=1}^{N}} \|DSX - T\|^2 + \lambda_{rank} \sum_{i=1}^{N} \alpha_i \|M_{i(i)}\|_{tr} + \lambda_{tv} TV(X), \text{ subject to } X_{(i)} = M_{i(i)}, i = 1, \ldots, N.$$

Alternating Direction Method of Multipliers (ADMM)

Convex Optimization Problem:

$$\min_{X,\{M_i\}_{i=1}^N} \|DSX - T\|^2 + \lambda_{rank} \sum_{i=1}^{N} \alpha_i \|M_{i(i)}\|_{tr}$$
$$+\lambda_{tv} TV(X),\ \text{subject to } X_{(i)} = M_{i(i)},\ i = 1, \ldots, N.$$

Similar to:

$$\min_{x \in \mathbb{R}^n} f_1(x_1) + f_2(x_2) \quad \text{subject to } A_1x_1 + A_2x_2 = b$$

How to solve :
(Objective Function)

$$\min_{x \in \mathbb{R}^n} f_1(x_1) + f_2(x_2) + \frac{\rho}{2}\|A_1x_1 + A_2x_2 - b\|_2^2$$
$$\text{subject to } A_1x_1 + A_2x_2 = b$$

Augmented :
Lagrangian

$$L_\rho(x_1, x_2, u) = f_1(x_1) + f_2(x_2) + u^T(A_1x_1 + A_2x_2 - b) + \frac{\rho}{2}\|A_1x_1 + A_2x_2 - b\|_2^2$$

http://www.stat.cmu.edu/~ryantibs/convexopt-F13/lectures/23-dual-meth.pdf

How does Low Rank Total-Variation work ?

Augmented Lagrangian based on ADMM (alternating direction method of multipliers:

$$\min_{X,\{M_i\}_{i=1}^N,\{Y_i\}_{i=1}^N} \|DSX - T^2\|$$
$$+ \lambda_{rank} \sum_{i=1}^{N} \alpha_i \|M_{i(i)}\|_{tr} + \lambda_{tv} TV\left(X\right) \quad + \sum_{i=1}^{N} U_i \left(X_{(i)} - M_{i(i)}\right) + \sum_{i=1}^{N} \frac{\rho}{2} \|X - M_i\|^2.$$

$$\min_{X,\{M_i\}_{i=1}^N,\{Y_i\}_{i=1}^N} \|DSX - T\|^2$$
$$+ \lambda_{rank} \sum_{i=1}^{N} \alpha_i \|M_{i(i)}\|_{tr} + \lambda_{tv} TV\left(X\right)$$
$$+ \sum_{i=1}^{N} \frac{\rho}{2} \left(\|X - M_i + Y_i\|^2 - \|Y_i\|^2\right).$$

How does Low Rank Total-Variation work? (2)

-> Min of X, M{i}, Y{i}

$$\min_{X,\{M_i\}_{i=1}^N,\{Y_i\}_{i=1}^N} \|DSX - T\|^2 + \lambda_{rank}\sum_{i=1}^{N}\alpha_i\|M_{i(i)}\|_{tr} + \lambda_{tv}TV(X) + \sum_{i=1}^{N}\frac{\rho}{2}\left(\|X - M_i + Y_i\|^2 - \|Y_i\|^2\right).$$

X: $$\arg\min_X \|DSX - T\|^2 + \lambda_{tv}TV(X) + \sum_{i=1}^{N}\frac{\rho}{2}\|X - M_i^{(k)} + Y_i^{(k)}\|^2$$

M{i}: $$\min_{\{M_i\}_{i=1}^N} \lambda_{rank}\sum_{i=1}^{N}\alpha_i\|M_{i(i)}\|_{tr} + \sum_{i=1}^{N}\frac{\rho}{2}\|X^{(k+1)} - M_i + Y_i^{(k)}\|^2$$

Y: $$Y_i^{(k+1)} = Y_i^{(k)} + \left(X^{(k+1)} - M_i^{(k+1)}\right).$$

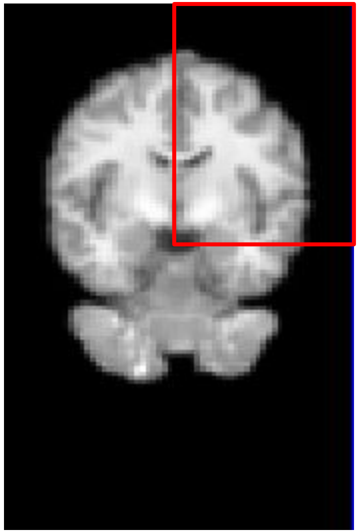
How does ADMM help/ Low Rank Total-Variation work? (Come, Intuition!)

- Bits form bites!
- Iterative computation
- Convergence

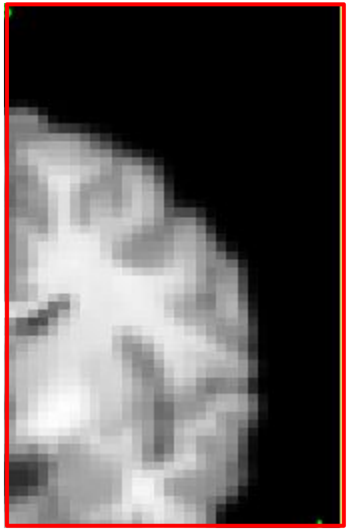
$$\min_{X,\{M_i\}_{i=1}^{N}} \|DSX - T\|^2 + \lambda_{rank} \sum_{i=1}^{N} \alpha_i \|M_{i(i)}\|_{tr} + \lambda_{tv} TV(X), \text{ subject to } X_{(i)} = M_{i(i)},\ i = 1, \dots, N.$$

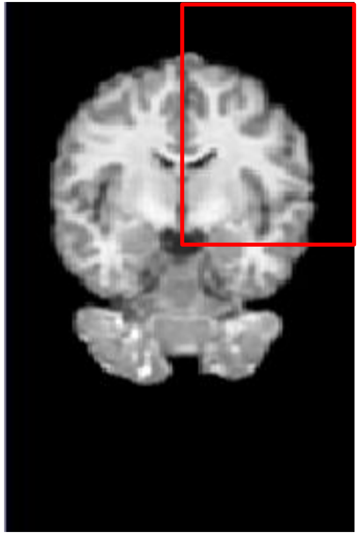
❖ Global minima and Edge preservation
❖ Redundancy reduction

Results

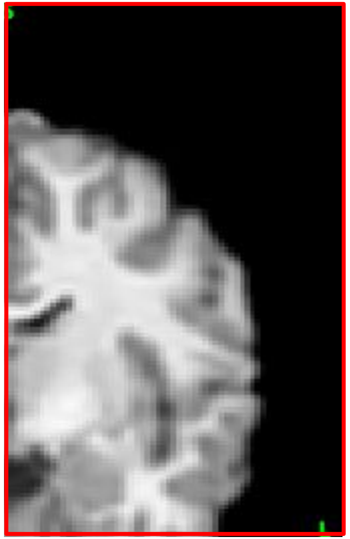


Input





Output



Results - Numbers

Iterations vs SNR

```
HaLRTC: iterations = 1   difference=0.663402
 snr= 11.325279
HaLRTC: iterations = 2   difference=0.512702
 snr= 17.129597
HaLRTC: iterations = 3   difference=0.132104
 snr= 18.361942
HaLRTC: iterations = 4   difference=0.043999
 snr= 18.711050
HaLRTC: iterations = 5   difference=0.022773
 snr= 18.902533
HaLRTC: iterations = 6   difference=0.015648
 snr= 19.034206
```

Img_1	Img_2	Peak_SNR	SSIM
Input	SpRes	51.729	0.95695
SpRes	HiRes	50.552	0.95659
Input	HiRes	47.417	0.91432

Peak SNR: Peak Signal to Noise Ratio
SSIM: Structural SIMilarity (SSIM) index

Limitations:

- Blur
- Run Time
- Tuning Parameters vary for T1 and T2.

Further Steps:

- Sucessfully implementing for T2
- Adding motion artifact removal


thank you