Nearly Linear-Time Algorithms for Structured Sparsity

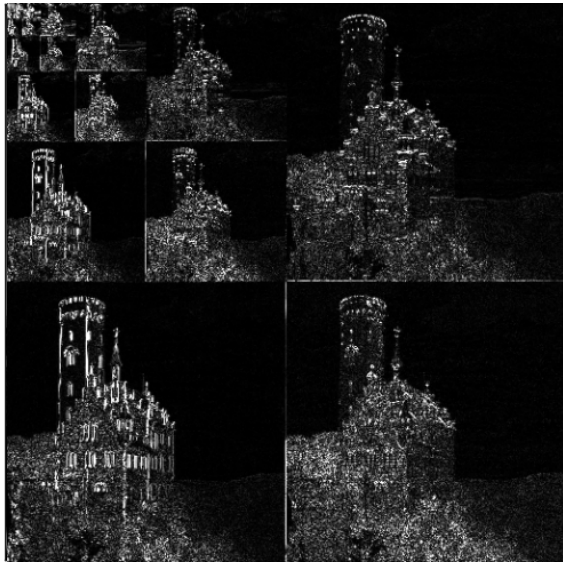
Chinmay Hegde
Piotr Indyk
Ludwig Schmidt
MIT

Compressive sensing

- Setup:
  - Data/signal in n-dimensional space : x
    E.g., x is an 256x256 image $\Rightarrow$ n=65536
  - Goal: compress x into Ax ,
    where A is a m x n "measurement" or "sketch" matrix,
    m << n
- Goal: want to recover an "approximation" x* of k-sparse x from Ax+e, i.e.,

$$||x^*-x|| \leq C \; ||e||$$

- Want:
  - Good compression (small m=m(k,n))
  - Efficient algorithms for encoding and recovery



Lichtenstein img processing test

$$\begin{pmatrix} & A & \end{pmatrix} \begin{pmatrix} x \end{pmatrix} = \begin{pmatrix} Ax \end{pmatrix}$$

Bounds

- Want:
  - Good compression (small $m=m(k,n)$)

    $m=O(k \log (n/k))$ [Candes-Romberg-Tao'04,….]

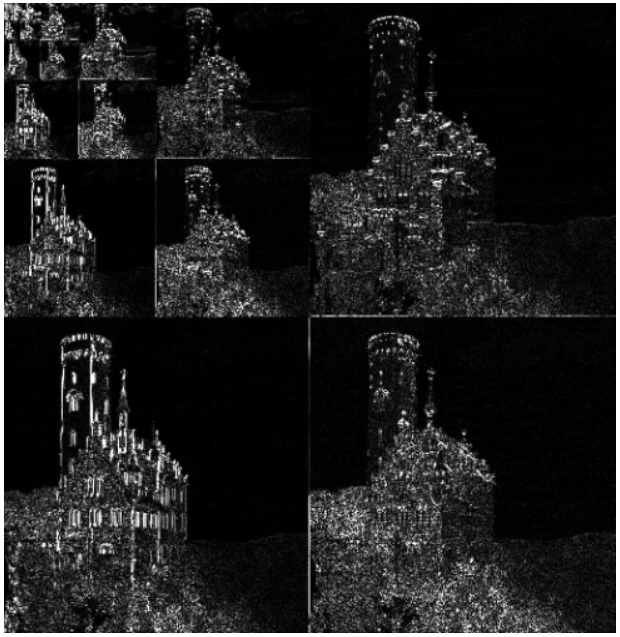
  - Efficient algorithms for encoding and recovery

    L1 minimization, CoSaMP, IHT, SMP, ….

- Issue ?
  - $\log (n/k)$ penalty compared to non-linear compression
  - Unavoidable in general [Donoho'04, Do Ba-Indyk-Price-Woodruff'10]

Structured sparsity

- Some signals contain more structure than mere sparsity
- Less sparsity patterns to worry about



Model-based compressive sensing

[Baraniuk-Cevher-Duarte-Hegde'10]

- Idea: structure $\Leftrightarrow$ restricted support
- Definition:

  A **structured sparsity model** M is defined by a set of allowed supports $\mathrm{M} = \{\Omega_1, \ldots, \Omega_{a_p}\}$ where $\Omega_i \subseteq [n]$:

$$M = \{x \in \mathrm{R}^n \mid \exists\, \Omega_i \in \Omega : \mathrm{supp}(x) \subseteq \Omega_i\}$$

- Only $O(k + \log |\mathrm{M}|)$ measurements suffice if $|\Omega_i| <= k$
- For all models considered in this talk $|\mathrm{M}|=2^{O(k)}$, so $m=O(k)$

Model specs

- Recovery algorithm depends on the model *M*
- Need **model-projection oracle**

$$M(x) = \mathrm{argmin}_{x' \in M} ||x'-x||_2$$

- Requirements:
  - The oracle should run **fast** (e.g. in polynomial or linear time)
  - The oracle must find **the best** approximation in the model

Necessary ???

- Model-IHT: iterate

$$x^i \leftarrow M(x^{i-1} + A^T(y - Ax^{i-1}))$$

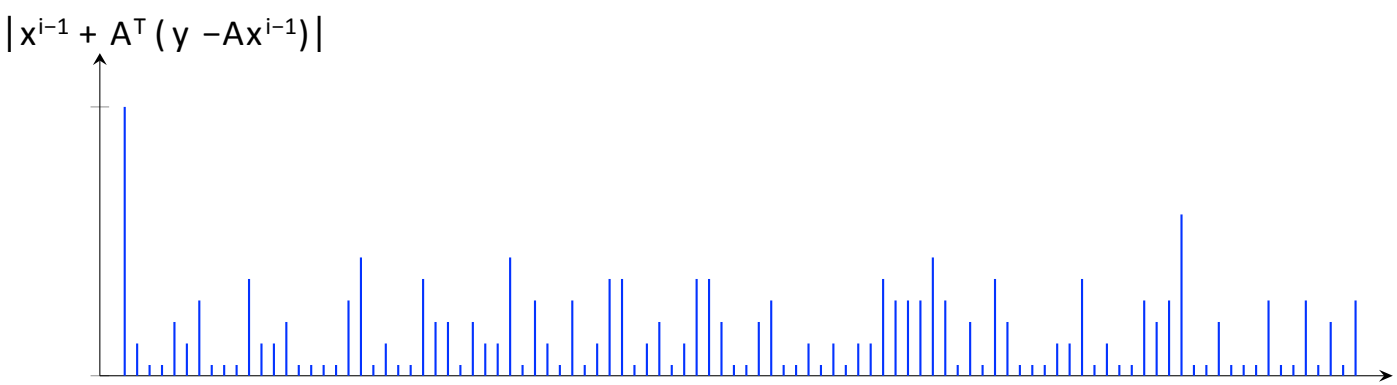
- Summary: If $|M|=2^{O(k)}$ and the oracle M() available, then Model-IHT recovers any x s.t. $\mathrm{supp}(x) \in M$ from O(k) measurements Ax
  - Stable generalizations exists as well
  - Theoretical and empirical improvement

Why not approximations?

- E.g., why not an approximate model-projection: given x find $x' \in M$ s.t.:

$$||x-x'||_2 \leq c \min_{x'' \in M} ||x''-x||_2$$

- Turns out MIHT might not work if c>1 !
  - While iterating $x^i \leftarrow M(x^{i-1} + A^T(y - Ax^{i-1}))$ we can keep $x^i=0$ even though the optimal solution is 1-sparse



Our framework [HIS14]

- Approximation-Tolerant Model-Based Compressive Sensing
- Intuition:
  - Consider oracle $M(x) = \mathrm{argmin}_{\Omega \in M} || x - x_\Omega ||_2$
    - **Minimizes** the norm of the **"tail"**
  - Equivalently: $M(x) = \mathrm{argmax}_{\Omega \in M} || x_\Omega ||_2$
    - **Maximizes** the norm of the **"head"**
  - However, these two problems are **not** equivalent if approximation is allowed
  - Our approach: required **two** separate approximate head and tail oracles…
  - …and then things work ☺

Our framework ctd.

- Tail-approximation oracle $T(x, p)$
  - $T(x, p) = x_\Omega$ for some $\Omega \in \mathrm{M}$
  - Tail approximation:

$$||x - T(x, p)||_2 \leq c_T \min_{\Omega \in \mathrm{M}} ||x - x_\Omega||_2$$

- Head-approximation oracle $H(x, p)$
  - $H(x, p) = x_\Omega$ for some $\Omega \in \mathrm{M}$
  - Head approximation:

$$||H(x, p)||_2 \geq c_H \max_{\Omega \in \mathrm{M}} ||x_\Omega||_2$$

- Approximation-Tolerant MIHT:

$$x^i \leftarrow T(x^{i-1} + H(A^T(y - Ax^{i-1}))$$

Approximation-Tolerant M-IHT

Theorem [HIS14]: The iterates of AM-IHT satisfy

$$\|x - x_{i+1}\|_2 \leq (1 + c_T)\left(\delta + \sqrt{1 - (c_H(1-\delta) - \delta)^2}\right)\|x - x_i\|_2$$

if *A* satisfies the model-RIP with constant δ and c_T and c_H are as on the previous slide.

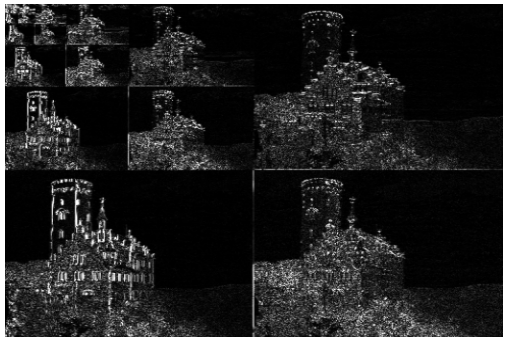
Prior work

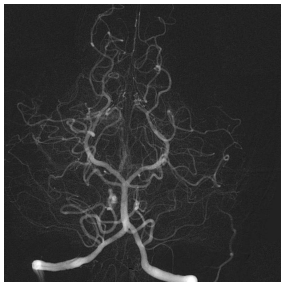
	Oracle	Stable Recovery	Assumptions
Blumensath	Additive	No	-
Kyrillidis-Cevher	Head	No	-
Giryes-Elad	Tail	Yes	Bound on singular values of *A*
Davenport-Needell-Wakin	Head, Tail	Yes	Portion of optimal support identified
This work	**Head, Tail**	**Yes**	**None**

[Giryes, Needell]: similar approach, somewhat different results

OK, but what does it lead to ?

- Constrained Earth-Mover Distance [SAMPTA'13, SODA'14]
  - Columns with similar sparsity patterns
  - Polytime recovery from O(k) measurements
- Tree-sparsity [ISIT'14, ICALP'14]
  - Supports form connected tree components
  - Near-linear time recovery from O(k) measurements
- Clustered sparsity [???'15]
  - Supports form connected components in graphs
  - Near-linear time recovery from O(k) measurements





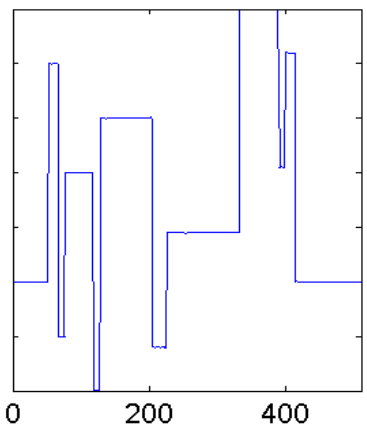
Tree-Sparsity

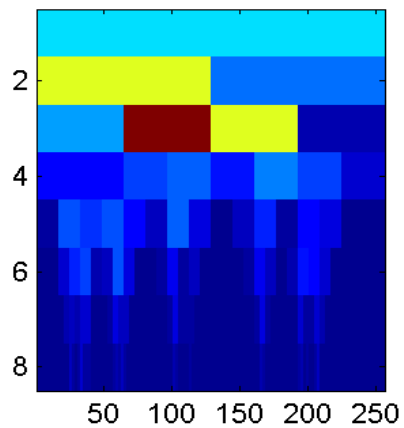
Tree-sparsity

- Sparse signals whose large coefficients can be arranged in the form of a ***rooted, connected tree***

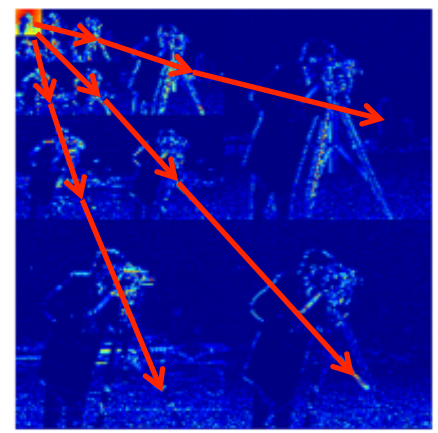
Applications: data streams, imaging, genomics, …

Piecewise const. signals





Natural images

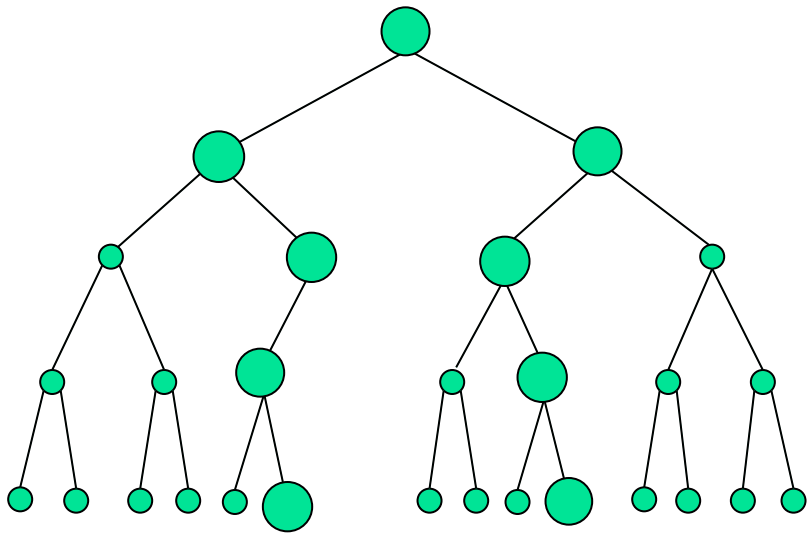


An optimization problem

- Given a signal x, compute the optimal (exact) **tree-sparse projection**

of x, i.e., solve

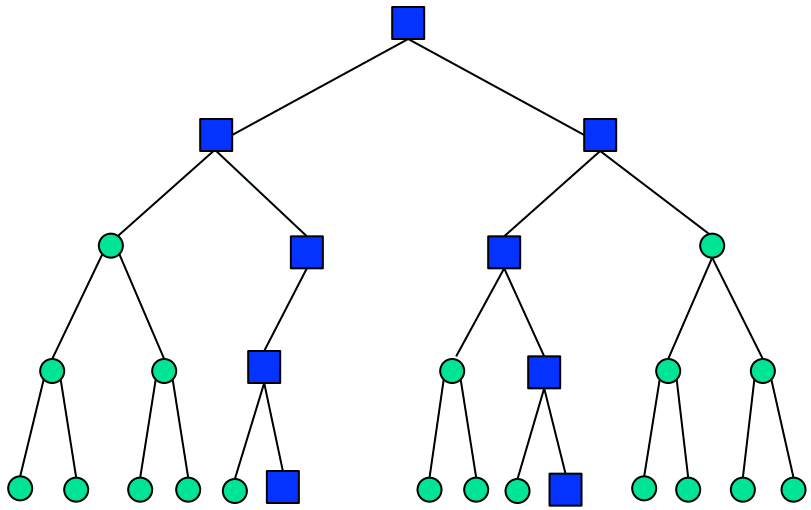
$$\min_{|\Omega| \leq k,\ \Omega \text{ tree}} ||x - x_\Omega||_2^2$$



An optimization problem

- Given a signal x, compute the optimal (exact) **tree-sparse projection** of x, i.e., solve

$$\min_{|\Omega| \leq k,\ \Omega \text{ tree}} ||x - x_\Omega||_2^2$$



Summary of results

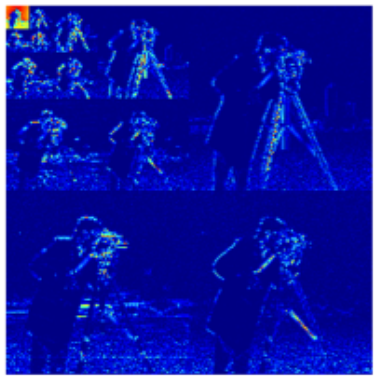
	Runtime	Guarantee
Baraniuk-Jones '94	O(n log n)	?
Donoho '97	O(n)	?
Bohanec-Bratko '94	$O(n^2)$	Exact
Cartis-Thompson '13	O(nk)	Exact
This work	O(n log n)	**Approximate Head**
This work	$O(n \log n + k \log^2 n)$	**Approximate Tail**

Implication: stable recovery of tree-sparse signals from O(k) measurements in time

#iterations * ($n \log n + k \log^2 n$ + matrix-vector-mult-time)

Why nlogn is better than nk

- Consider a 'moderate' problem size
  - e.g. $n = 10^6$, k = 5% of n



- Then, nk ~ **50×10^9** while nlog n ~ 20×10^6

- Really need near-linear time

Tail Approximation

- We want to approximate:

$$\min_{|\Omega|\leq k,\ \Omega \text{ tree}} ||x-x_\Omega||_2^2$$

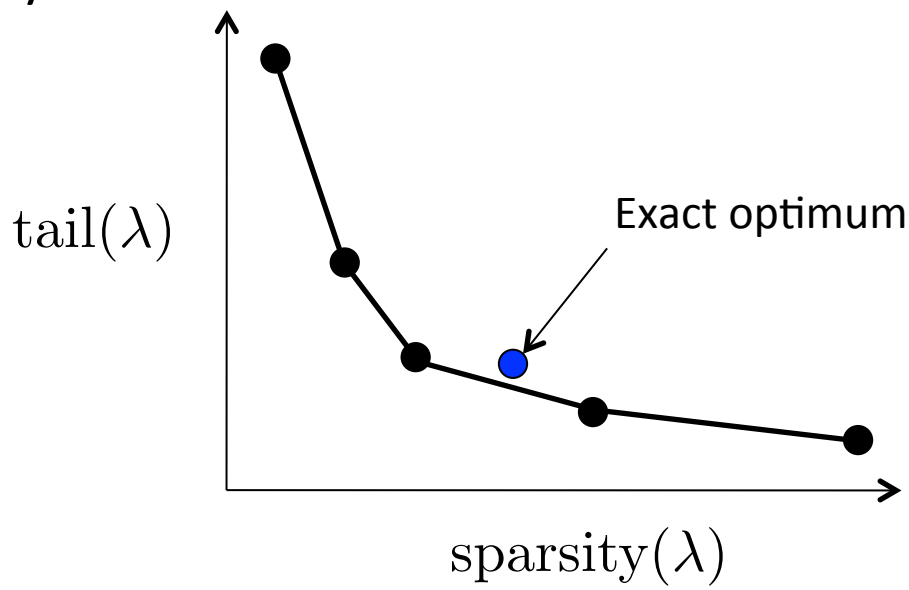
- Perform a Lagrange relaxation of the sparsity constraint:

$$\min_{\Omega \text{ tree}} ||x-x_\Omega||_2^2+\lambda|\Omega|$$

- Can be solved using a simple dynamic program (DP) using $O(n)$ time, $O(n)$ space
  - Similar approach as [Donoho '97]

How to choose λ ?

- Pareto analysis: examine the solution curve as a function of



- Via convexity of the curve, we can always obtain
  - Either: (≤2k, ≤OPT)
  - Or: (≤k, ≤2OPT)
- Altogether: we get a tail guarantee in time roughly O(n log n)

Head approximation

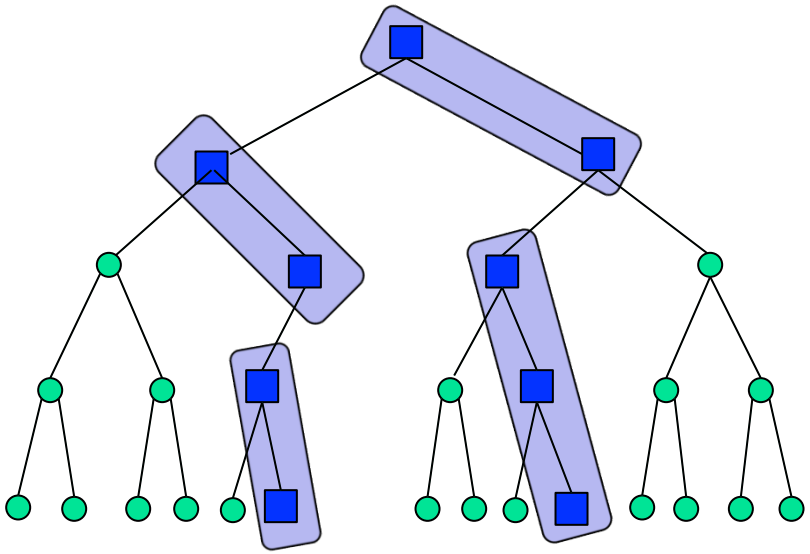
- Want to approximate:

  $\max_{|\Omega| \le k,\ \Omega \text{ tree}} ||x_\Omega||_2^2$

- Lemma: The *optimal* tree-projection can be always decomposed into **disjoint** pieces of size O(log n)

- **Greedy** approach: compute exact projections with sparsity O(log n), assemble pieces, merge
- Total running time:

  $O(k \log^2 n + n \log n)$



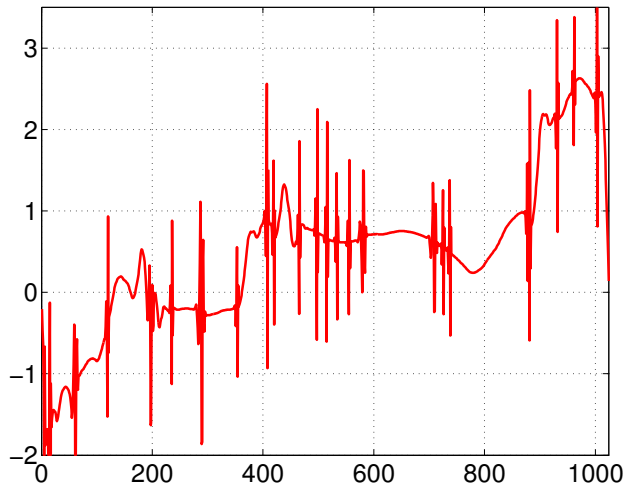
Experiments: 1D signals

n = 1024
k = 40
m= 140

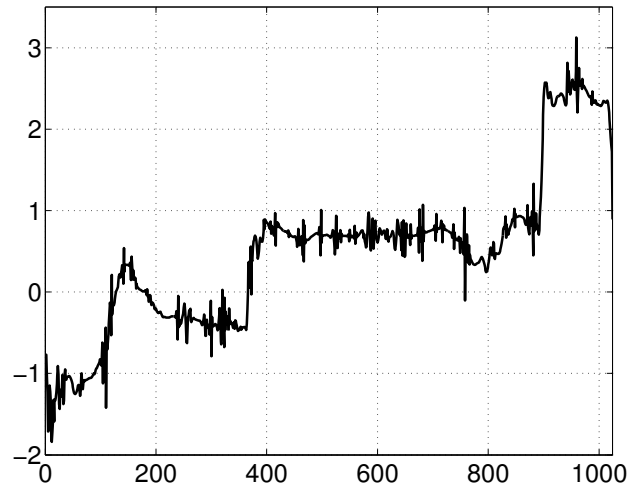
Gaussian measurements

Original signal x is tree-sparse in the **wavelet basis**

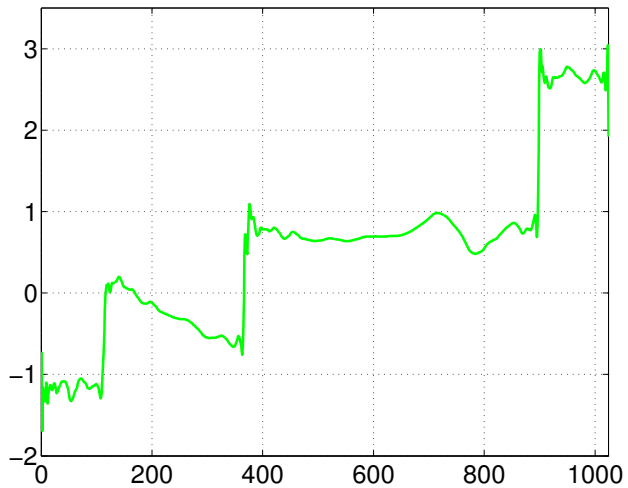
Both Tree-CS and AM-CoSaMP exactly recover signal



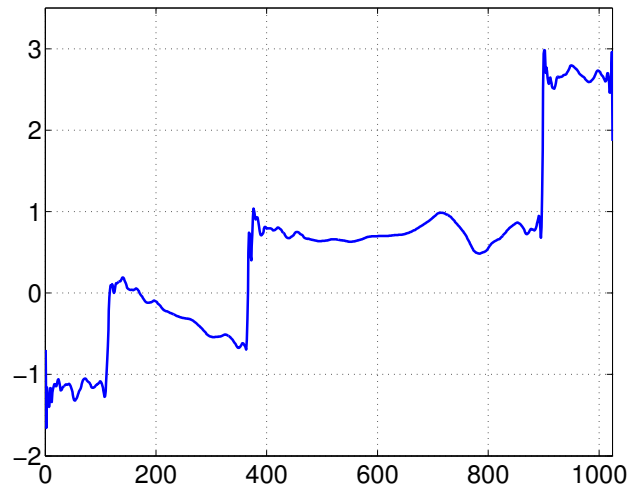
CoSaMP



SPGL1



Tree-CoSaMP, Exact

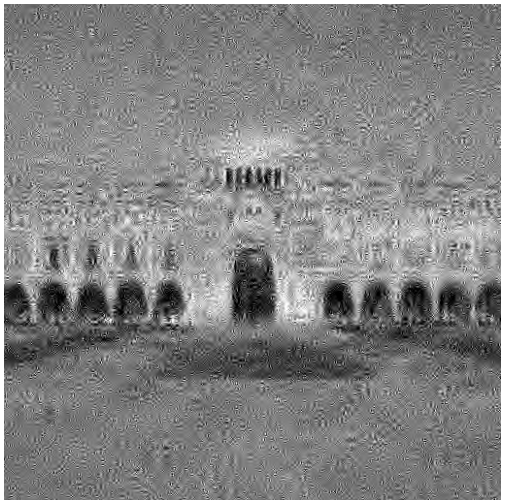


AM-CoSaMP

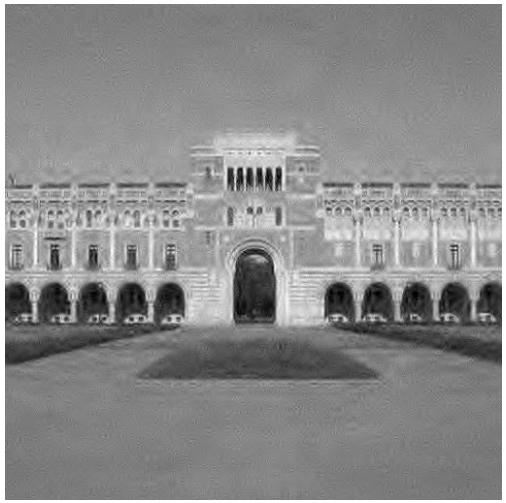
Experiments: 2D images

n = 512 x 512
k ~ 10,000
m ~ 35,000

Fourier measurements



CoSaMP



SPGL1

Tree-CoSaMP, Exact

AM-CoSaMP

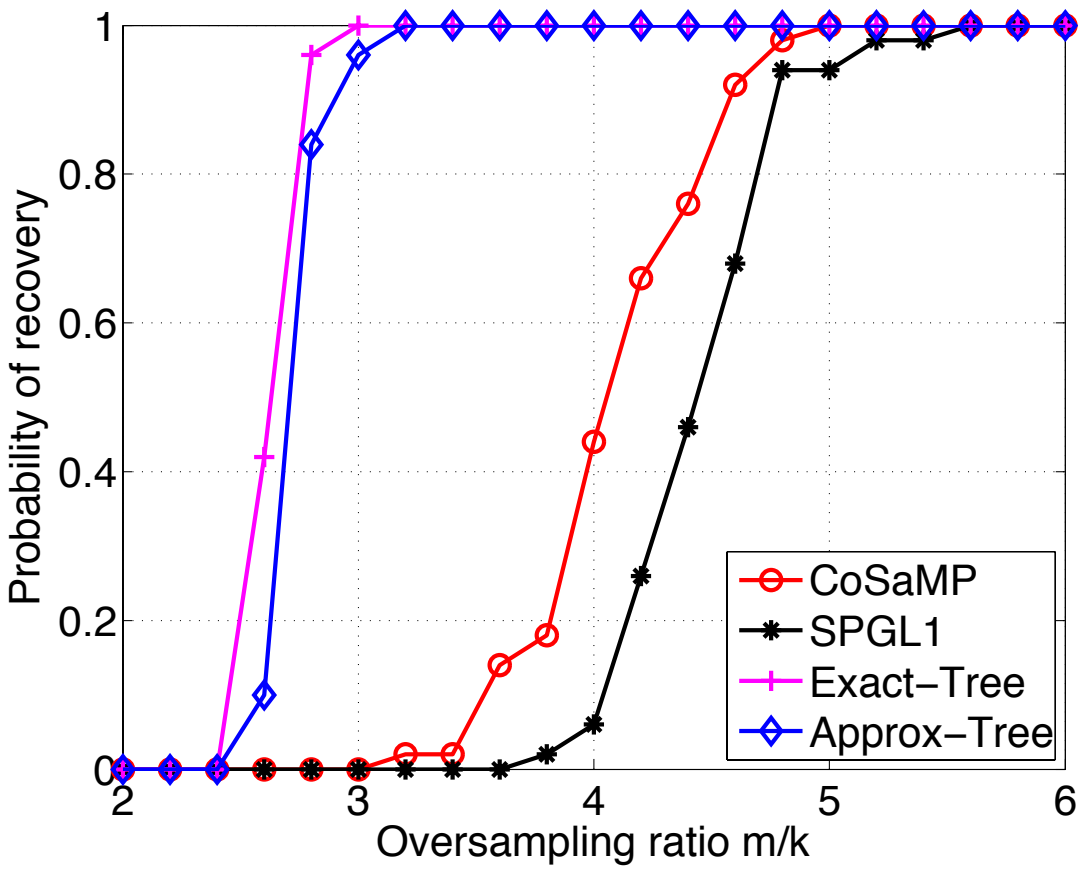
Image by AniRaptor2001

Experiments: Monte Carlo

Test signals:
piecewise polynomial
(tree-sparse in the **wavelet basis**)

Trials randomized over choice of measurement matrix

Successful recovery :
Error is within 5% of the norm of the original signal



Experiments: Speed

Algorithm	**Exact**	**TailApprox**	**2 Matlab FFTs**
Runtime	13.54 sec	0.0161 sec	0.0136 sec

TailApprox offers ***840x*** *speedup* over the exact-tree projection method

(In fact, runtime comparable to Matlab's FFT)

Code available at:
http://people.csail.mit.edu/ludwigs/code.html

Conclusions

- Can make model-based compressive sensing approximation tolerant
- This lets us use approximation algorithms tools to make recovery tractable/faster
  - Network flows, Dynamic programming, Greedy, Steiner tree,…
- Open questions:
  - Other models?