Robust detection of oscillations in SDO/AIA solar data

Rebecca Willett

Duke University, Electrical and Computer Engineering

February 17, 2012

Observation model

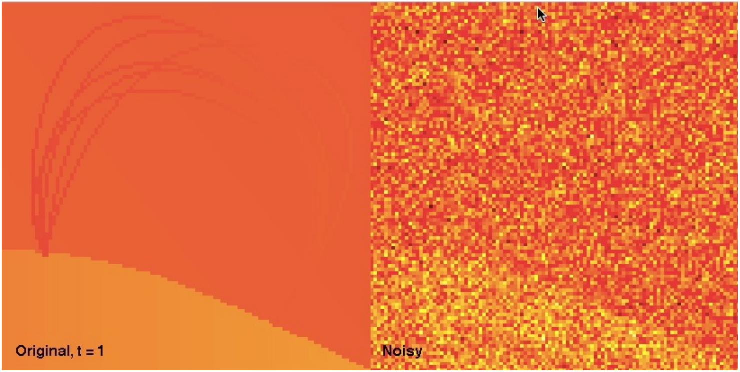
At pixel i, time t, we have signal

$$d_{i,t} = A_i \cos(\omega_i t) + B_i \sin(\omega_i t)$$

and observation

$$y_{i,t} = d_{i,t} + \underbrace{n_{i,t}}_{noise}$$

(This model doesn't account for multiple wavebands.)



Current approach

Very broadly speaking, the current approach has these steps:

1. for each pixel i
   - compute Fourier transform of $\mathbf{y}_i = [y_{i,1}, \ldots, y_{i,t}, \ldots, y_{i,T}]^T$
   - calculate probability

$$p_i \triangleq \mathbb{P}\left(\omega_i \neq 0\right)$$

2. form **probability map** $\mathbf{p} = [p_i]$
3. perform spatial smoothing (boxcar, median, wavelet, curvelet, etc) on $\mathbf{p}$.

Key assumptions

The success of this approach depends on two key assumptions:

1. each pixel has only a small number of dominant frequencies (ideally one)
2. the probability map **p** varies smoothly across space

Chief difficulty

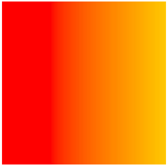
Accurately estimating **p** is a hard, especially because spatial information isn't used until final step.

Proposed new assumptions

Can we do better if we make slightly different assumptions? And are these assumptions consistent with the physical reality?

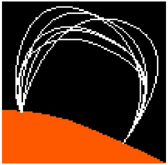
New assumptions

1. each pixel has only a small number of dominant frequencies (ideally one)
2. A_i, B_i and ω_i all vary piecewise-smoothly across space
3. oscillations lie in a low-dimensional subspace (*i.e.*, there are a few representative oscillations, and all true oscillations are a weighted combination of those few)



A

B



ω

Toolbox

Here are some tools designed to exploit these data satisfying these assumptions:

1. Sparse estimation via coefficient thresholding
2. Yaroslavsky's filter for spatial smoothing
3. Principle components analysis for low-rank video approximation

Sparse estimation

We observe the time series

$$\mathbf{y}_i = \mathbf{d}_i + \mathbf{n}_i$$

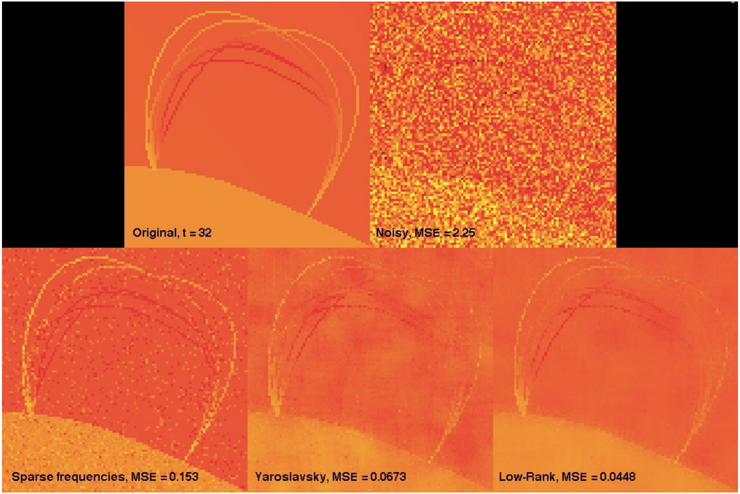
and assume $\mathbf{d}_i$ is sparse in a Fourier basis. We can use the sparsity assumption to estimate $\mathbf{d}_i$ from $\mathbf{y}_i$:

$$\begin{aligned} \widehat{\boldsymbol{\theta}}_i &= \text{argmin}_{\boldsymbol{\theta}} \|\boldsymbol{\theta} - \text{DFT}(\mathbf{y}_i)\|_2^2 + \tau \|\boldsymbol{\theta}\|_1 \\ &= \text{SoftThreshold}_\tau(\text{DFT}(\mathbf{y}_i)) \\ \widehat{\mathbf{d}}_i^{FT} &= \text{IDFT}(\widehat{\boldsymbol{\theta}}_i) \end{aligned}$$

This approach finds the estimate $\widehat{\mathbf{d}}_i^{FT}$ which is

- close to the data $\mathbf{y}_i$
- sparse in the Fourier domain

Example


Original, t = 32
Noisy, MSE = 2.25
Sparse frequencies, MSE = 0.153
Yaroslavsky, MSE = 0.0673
Low-Rank, MSE = 0.0448

Kernel-based image denoising: $\widehat{d}_{i,t} = \sum_j w_{i,j} y_{j,t}$

Usual kernel method [a]

$$w_{i,j} = K_h(x_i, x_j)$$

- w has no dependency on y
- K: kernel and h: bandwidth (smoothing parameter)
- Gaussian kernel example : $K_h(x_i, x_j) = e^{-\|x_i - x_j\|_2^2 / 2h^2}$

[a] Nadaraya '64, Watson '64

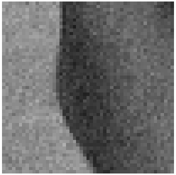
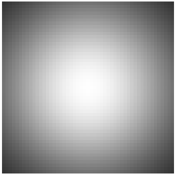


Image Search Zone



Spatial

Kernel-based image denoising: $\widehat{d}_{i,t} = \sum_j w_{i,j} y_{j,t}$

Yaroslavsky/Bilateral Filter [a]

$$w_{i,j} = K_h(x_i, x_j) L_{h_y}(y_{i,t}, y_{j,t})$$

- Use spatial *and* photometric proximity
- K, L: kernels; h, h_y: bandwidths (smoothing parameters)

[a] Yaroslavsky '85, Lee '83, Tomasi and Manduchi '98

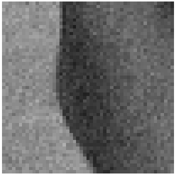
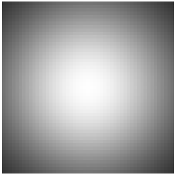
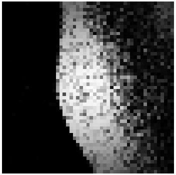


Image Search Zone



Spatial



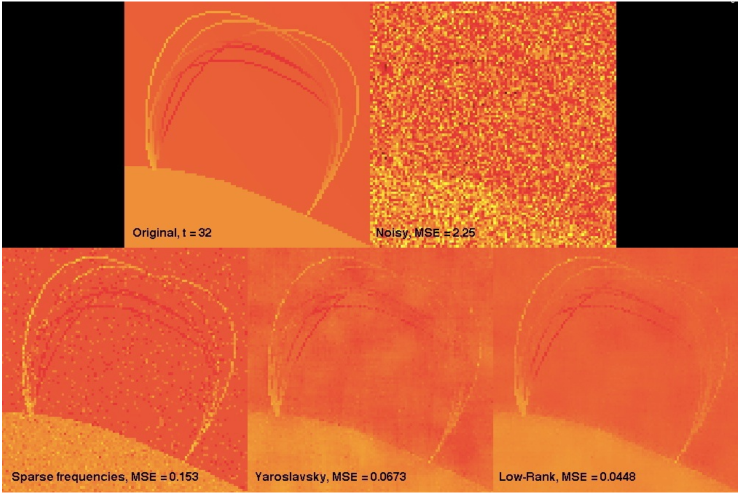
Yaroslavsky / Bilateral

Yaroslavsky's filter for video data

In our setup, we don't have a single image, but rather a time-series of images. Thus, instead of having our filter depend on individual pixel similarities, we will have it depend on **time-series similarities**:

$$\widehat{d}_{i,t}^{YF} = \sum_j d_{j,t}^{FT} \underbrace{K_h(x_i, x_j) L_{h_y}(\mathbf{y}_i, \mathbf{y}_j)}_{\text{weight independent of time, based on entire time series}}$$

Example



Original, t = 32
Noisy, MSE = 2.25
Sparse frequencies, MSE = 0.153
Yaroslavsky, MSE = 0.0673
Low-Rank, MSE = 0.0448

PCA

Finally, we want to use the fact that there are a few representative oscillations in the data, and all observed oscillations are a weighted combination of these representatives. To do this, form the matrix

$$\widehat{D}^{YF} = [\mathbf{d}_1^{YF}\, \mathbf{d}_2^{YF}\ \cdots, \mathbf{d}_N^{YF}]$$

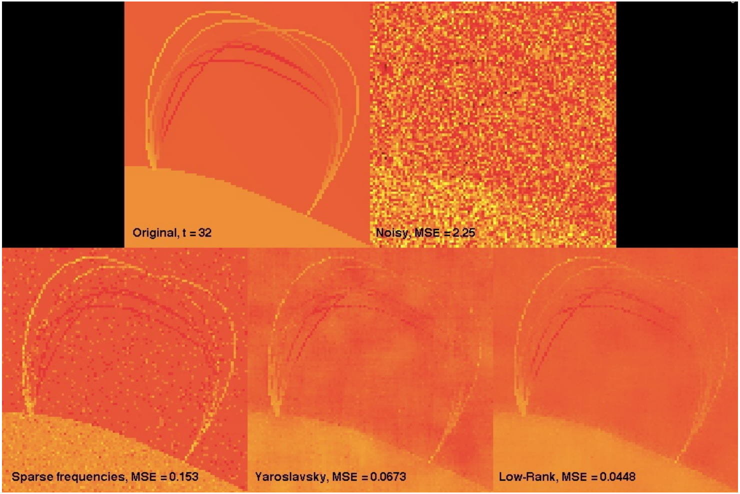
so each column corresponds to a time series in a different pixel. Next compute the SVD of $\widehat{D}^{YF}$:

$$\widehat{D}^{YF} = USV^T$$

and only keep the largest elements of the diagonal matrix S to form $\widehat{S}$. Next let

$$\widehat{D}^{LR} = U\widehat{S}V^T.$$

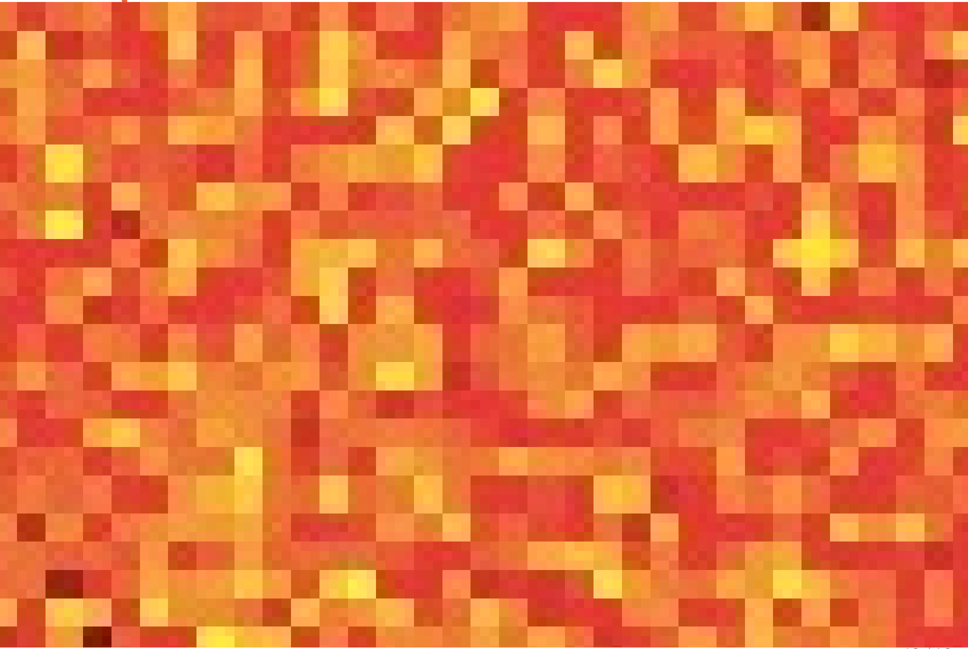
Example



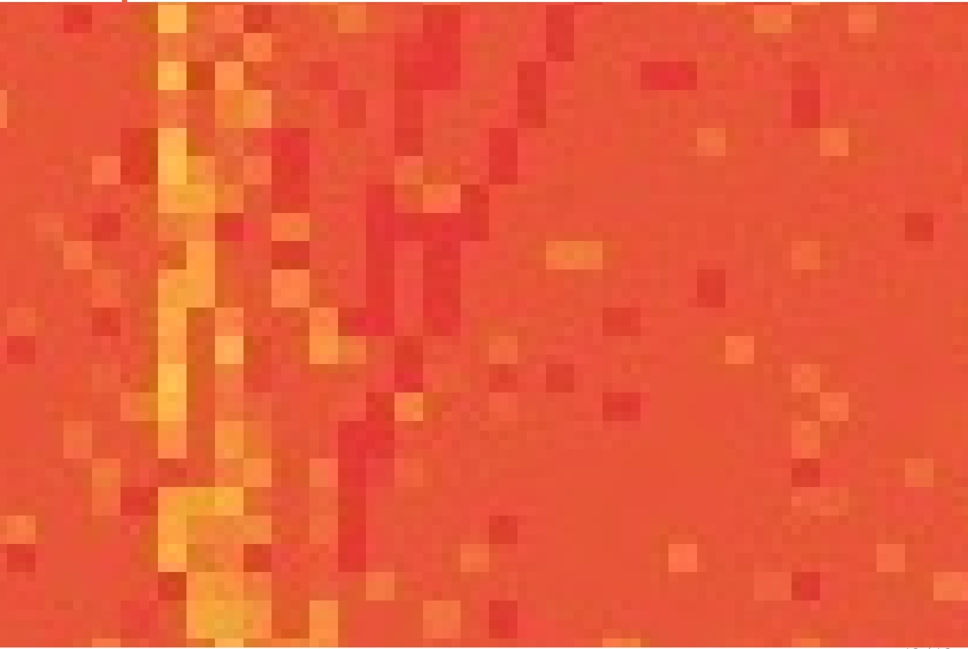
Original, t = 32
Noisy, MSE = 2.25
Sparse frequencies, MSE = 0.153
Yaroslavsky, MSE = 0.0673
Low-Rank, MSE = 0.0448

Example

Example



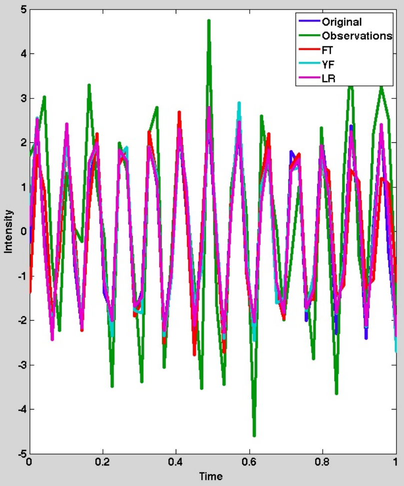
Example



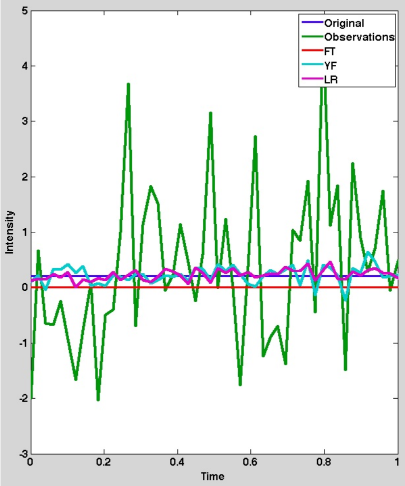
Example

Example

Example (cont.)



Oscillating pixel



Non-oscillating pixel

Discussion

- These tools can be used to pre-process data and improve the robustness and accuracy of oscillation detection
- Patch-based versions of these methods exist
  - can be used to reduce computational complexity or relax assumptions
  - can be used to better exploit underlying physical structure
- Performance will ultimately depend on how realistic the underlying assumptions are for real data
- I applied these tools sequentially; the optimal order or **joint** spatio-temporal reconstruction is an open problem
- All techniques described here can be applied to multiple wavebands simultaneously.
- At their heart, all these methods exploit **low-dimensional structure** (sparsity, low rank, piecewise smoothness) in the underlying high-dimensional observation space

Thank you.