# Compressed Sensing in Astronomy

**Hugo Dupré** **Philippe Peroumal**

March 27, 2017

## Contents

# 1 Introduction

Recent advances in signal processing have focused on the use of sparse representations in various applications. Compressed sensing is a new sampling framework that provides an alternative to the Shannon sampling theory. Compressed sensing is a technique to sample compressible signals below the Nyquist rate, while still allowing near optimal reconstruction of the signal. It can provide new insights into astronomical data compression. In this work, we refer to the following paper : https://arxiv.org/pdf/0802.0131.pdf ([2]). This paper by Bobin et Al. describes how compressed sensing is relevant to solve astronomical imagery problems.

In this report, we will study different compressed sensing methods for astronomical data reconstruction from sampled measurements. We will present the Isotropic Undecimated Wavelet transform, or Starlet transform, which we will use when studying images of galaxies. Then we will present the algorithms that we coded for denoising and reconstructing images in Astronomy. Finally we will setup and experiment : we will compare and study the algorithms' results on real data.

# 2 The starlet transform

We suppose that the signal we received is sparse in a particular basis. Our goal is to find an appropriate basis. The wavelet transform boils down to successively decompose the image into the coarse and the details. The relationship between the successive approximations are given by convolutions with suitable filters. [5] propose to use an isotropic undecimated wavelet transform : the starlet transform. The starlet transform has the following properties :

- the filters are symmetric
- the filters are isotropics
- the filters are separables : we can apply the filters to the row and the columns.

The two first properties match with the nature of our images. The last property provides faster calculus. We now provide the key-equations. The mother-wavelet $\psi$ is the function that once rescaled and dilated will give us the basis for each resolution. The mother-wavelet is scaled by the scaling function $\phi$.

As the filters are separable, we can consider first the scaling function in one dimension $\phi_{1D}$. It is a spline. We have the following relationships :

- $\phi_{1D}(t) = \frac{1}{12}[|t-2|^3 - 4|t-1|^3 + 6|t|^3 - 4|t+1|^3 + |t+2|^3]$
- $\phi(t_1, t_2) = \phi_{1D}(t_1)\phi_{1D}(t_2)$     It is the separability.
- $\frac{1}{4}\psi(\frac{t_1}{2}, \frac{t_2}{2}) = \phi(t_1, t_2) - \frac{1}{4}\phi(\frac{t_1}{2}, \frac{t_2}{2})$

The tremendous interest of the is its simplicity. Due to the last equation, the application of the starlet transform is simple. The low-pass filter $h_{1D}$ that gives the relation-ship between a resolution $j$ and a lowest resolution $j$ is just the convolution : $\frac{1}{16}[1, 3, 3, 1]$. Consequently, we first padd the image, then we apply $h_{1D}$ to each row, then we apply $h_{1D}$ to each column. We substract the result to obtain the details.

We have implemented this method only with standard libraries of `Python` : `numpy` and `pandas`, without any built-in signal processing library.

# 3 Algorithms for denoising and reconstructing images in Astronomy

The starlet transform is the foundation of our algorithms. In order to reconstruct the signal, we assume that the signal is sparse in this wavelet domain. We decompose the observation in this wavelet domain and at each scale, we apply an operator to remove the noise. We first present some tools.

## 3.1 A first inverse problem

We are dealing with an inverse problem : $b = Ax + n$

- $b$ is the observation
- $A$ is the observation operator, for the noise removing, $A = Id$
- $n$ is the noise. We suppose that the noise is additive and gaussian with variance $\sigma^2$

We use the soft-thresholding operator $S_\lambda$ and the thresholding operators $H_{\sqrt{\lambda}}$.They have closed form.

$$\underset{x}{\operatorname{argmin}} \lambda \|x\|_1 + \frac{1}{2} \|b - x\|_2^2 = S_\lambda(x)$$

$$\underset{x}{\operatorname{argmin}} \lambda \|x\|_0 + \frac{1}{2} \|b - x\|_2^2 = H_{\sqrt{\lambda}}(x)$$

We use these operators in the following context :

- $\sigma^2$ the variance of the noise is unknown : we need a robust estimator.
- $\Phi$ is the dictionnary ( the starlet wavelet here) in which the signal $x$ is sparse

### 3.2 Iterative Soft and Hard-Thresholding

The iterative Soft-Thresholding is presented in [3]. If the wavelets form an orthogonal basis, it boils down to solve :

$$\underset{x=\Phi\alpha}{\operatorname{argmin}} \lambda \|x\|_1 + \frac{1}{2} \left\|\Phi^T b - \alpha\right\|_2^2$$

In this case, we obtain a closed form :

$$\hat{x} = \Phi S_\lambda(\Phi^T b)$$

In the dictionnary domain, we threshold the smallest coefficients and then get back to the pixel domain of our astronomical image. We need to choose the threshold, and therefore the penalty. The variance of the noise is unknwown. A robust estimator is the median absolute deviation

$$\sigma_{MAD} = \rho.\text{median}(|\Phi^T b - \text{median}(\Phi^T b)|) \qquad \rho = 1.4826$$

So we sort the coefficients of the wavelets, and retain only the bigger accordingly to a false rate detection. It is a quantile of the gaussian law of the noise, whose estimated variance is given by the MAD.

For the hard-thresholding, we use the same steps :

- We go in the dictionnary domain.
- We estimate the level of noise with median absolute deviation for each scale.
- We deduce a threshold of false rate detection for each scale.
- We apply the hard-threshold operator to the coefficients of each scale.
- We get back to the pixel domain.

However in our case, the dictionnary, the starlet basis is not orthogonal. We need to use iterative methods. We still need to solve the following objective :

$$\underset{x=\Phi\alpha}{\operatorname{argmin}} \lambda \|x\|_1 + \frac{1}{2} \left\|\Phi^T b - \alpha\right\|_2^2$$

We use a forward-backward splitting algorithm. Fist, we operate a gradient descent on the second term, the data fidelity. Then we enforce the sparsity constraint by applying the proximal operator of the $L^1$-norm (the soft-thresholding operator) to the intermediary result. Even if the starlet transform is not orthogonal, we converge toward a local mimnimum.

For the iterative hard thresholding algorithm, we use the hard-thresholding operator in the algorithm. The iterative hard thresholding algorithm [1], when applied to the compressed sensing recovery

**Algorithm 1** Iterative soft-thresholding

**Input :** $b$ noisy observations, $\Phi$ dictionnary, number of steps $T$, $\gamma$ gradient-descent step
**Output :** $\alpha$ denoised decomposition in the dictionnary

1: $\alpha_0$ apply the starlet transform to $b$
2: obtain the penalty $\lambda$ with respecy to the Median-Absolute-Deviation (MAD)
3: **for** $i = 1$ to $T$ **do**
4: $\quad \beta \leftarrow \alpha^{(t)} + \gamma \Phi^T (b - \Phi\alpha)$
5: $\quad \alpha^{(t+1)} \leftarrow S_{\gamma\lambda}(\beta)$
6: **end for**

problem, verifies a bunch of properties : it gives near-optimal error guarantees, it is robust to observation noise, it succeeds with a minimum number of observations, the memory requirement is linear in the problem size and it requires a fixed number of iterations depending only on the logarithm of a form of signal to noise ratio of the signal.

We have implemented the hard and soft basic thresholding.

### 3.3 Iterative multiresolution mask denoising algorithm

If we threshold the coefficients of the starlet at each scale, we don't take into account the relationship between each scale. Indeed the basis is not orthogonal and therefore there is some redundancy. The redundancy is linked to the structure of the image. The iterative multiresolution mask and denoising algorithm [4] exploits this idea.

We can use an iterative algorithm to extract the structure of the image.

**Algorithm 2** Iterative multiresolution mask denoising algorithm

**Input :** $b$ noisy observations, $\Phi$ dictionnary, number of steps $T$
**Output :** $x^+$ structure, $x^-$ residuals

1: Apply the starlet transform to $b$
2: Compute the penalty $\lambda$ for the thresholding with respecy to the Median-Absolute-Deviation (MAD) for each scale.
3: The multimask resolution $M$ is the indicator function of the relevant coefficients.
4: **for** $i = 1$ to $T$ **do**
5: $\quad x^+ = x^- + \Phi M \Phi^T (b - x^-)$
6: **end for**

We have implemented this method with the standard libraries of `Python` : `Pandas` and `Numpy`.

## 4 Experiments

### 4.1 Setup and motivation

This experiment was done in Python 2.7, and the code is available here : `https://github.com/Caselles/Compressed_sensing_astronomy`.

In this experiment, we will use several infared galaxy images in order to test our methods. We will evaluate these methods on two tasks : denoising and reconstructing images. We will assess the different results using a metric. Let $x \in \mathbb{R}^n$ be the ground truth signal, and let $\widehat{x} \in \mathbb{R}^n$ be the estimated signal. We define the loss $l(x, \widehat{x})$, or error, of $\widehat{x}$ with respect to $x$, using the Froebenius norm :

$$l(x, \widehat{x}) = \|x - \widehat{x}\|_F = \sqrt{\sum_{j=1}^{n} \sum_{i=1}^{n} (x_{i,j} - \widehat{x}_{i,j})^2}$$

We will compare the algorithm's performances on different level of noise and different level of sparsity.

### 4.2 Results

First, let us see what type of images we will study. For this report, we will focus on one image. For comparison, the noisy version is displayed. The added noise is simply a random noise multiplied by a coefficient. In python, it would simply be : `300 * np.random.randn(256, 256)`. Let's have a look to the ground truth and noisy images.

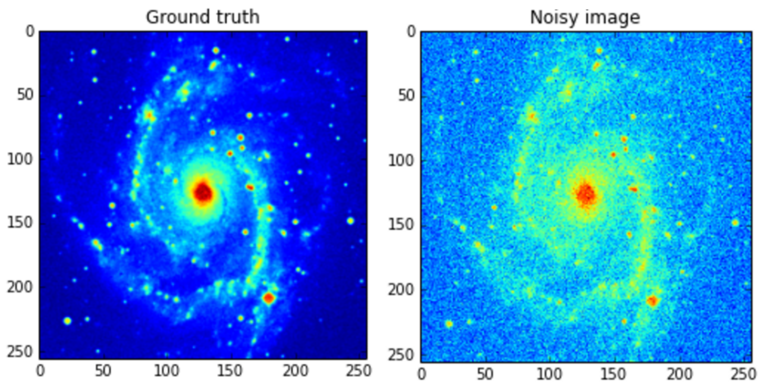


Figure 1: Ground truth and noisy image

As we can see, the noisy image is pretty different from the ground truth one. We would like to denoise this image with the three denoising algorithms we already presented : Soft-Thresholding, Hard-Thresholding and the Multiresolution mask denoising algorithm. Let's have a look at the results.

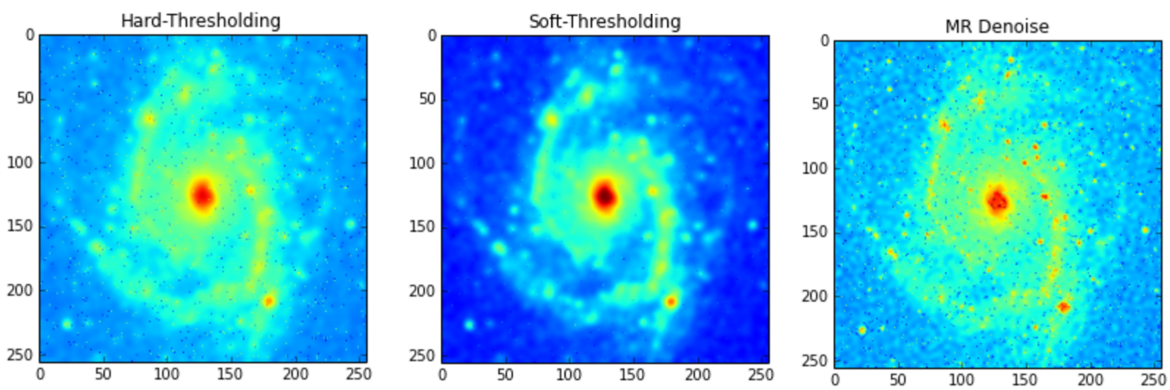


Figure 2: Results of the 3 algorithms

The results seem better with the Soft-Thresholding method. But let's have a look at the results if we change the noise level to only 50 :

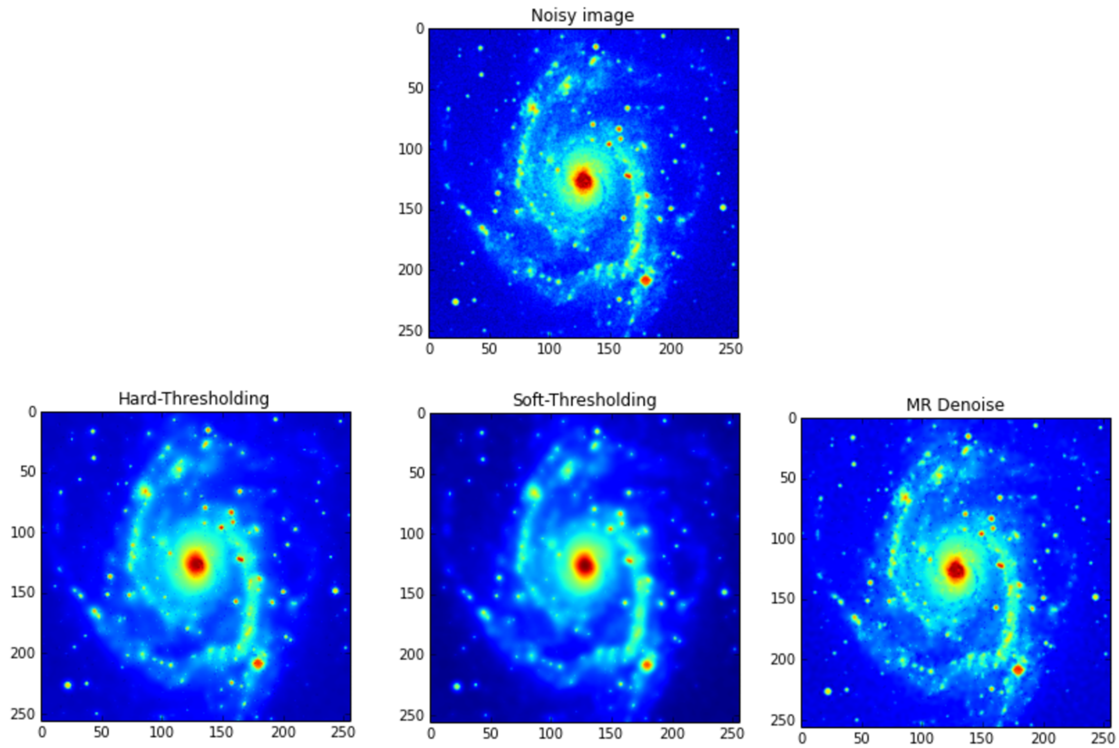


Figure 3: Error as a function of the sparsity

Now, the MR Denoise method seems to work best ! We need to evaluate the result more rigorously in order to assess the performance of the methods. We will use the Frobenius norm, and see how the algorithms behave on both the denoising task and the reconstructing task. Let's begin with the denoising task.

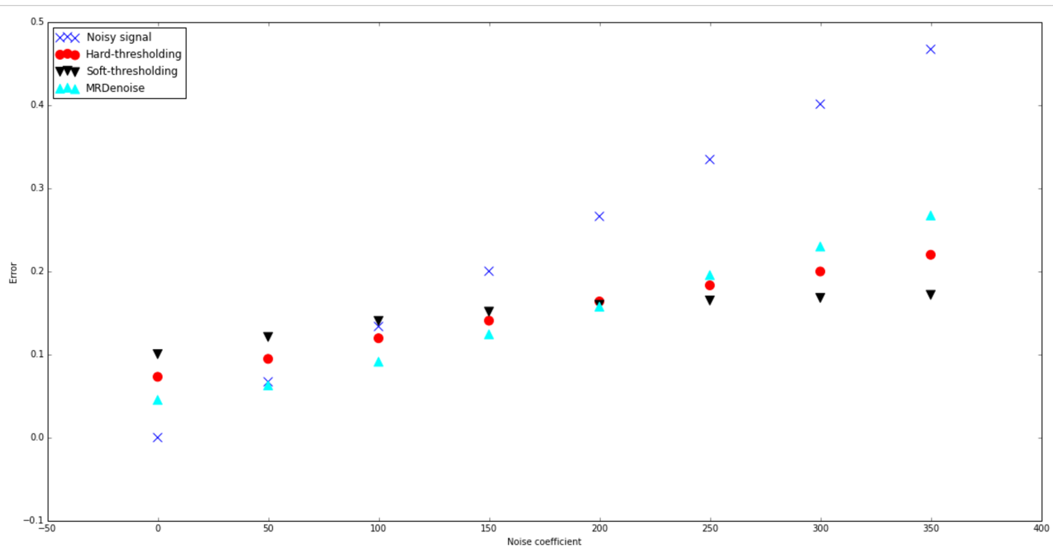


Figure 4: Error as a function of the noise

We can see that the algorithms seem to work for denoising images. The error is significantly lower than the sampled signal for each of the four algorithms, as long as the noise is not too small. If the noise is too small (coefficient of 50 or 100), we can see that the soft-thresholding and the hard-thresholding methods yield larger errors than the initial signal. When the noise gets larger, these methods seem to work very well. On the contrary, the MR Denoise method seem to work best for little noises and a bit worse for larger noises. Hence, one can choose the right method according the level of noise in the image to denoise. Now we will focus on the reconstructing task.

Let's first give an illustration of the results obtained with each of the method.

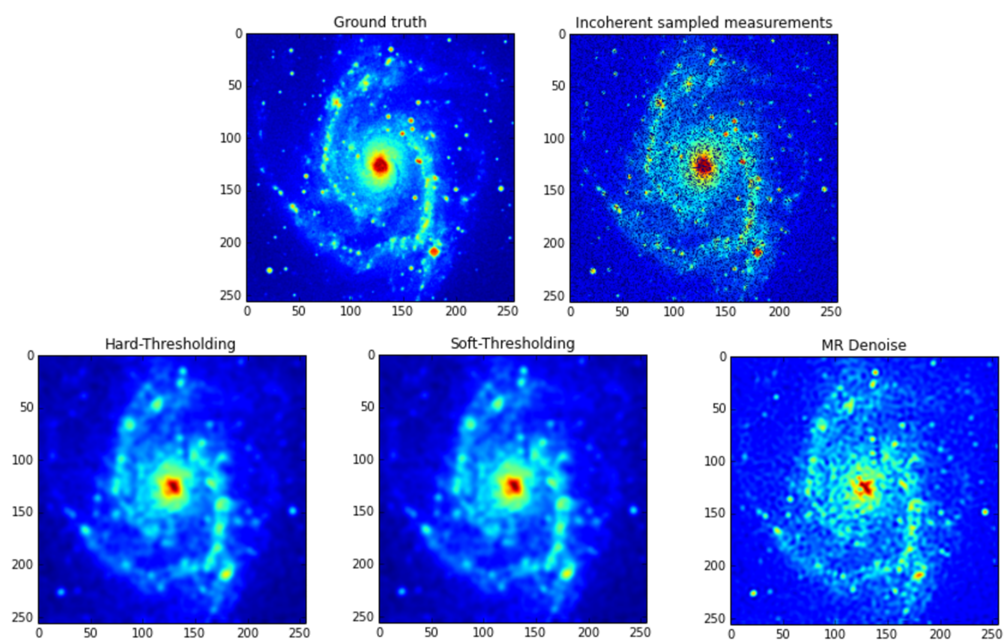


Figure 5: Results of the reconstructing task using the 3 algorithms.

As we expected, the images are reconstructed, and we can see the structure of the galaxy. Hard-thresholding and soft-thresholding seem to work better. We will study that through the use of the Frobenius norm.

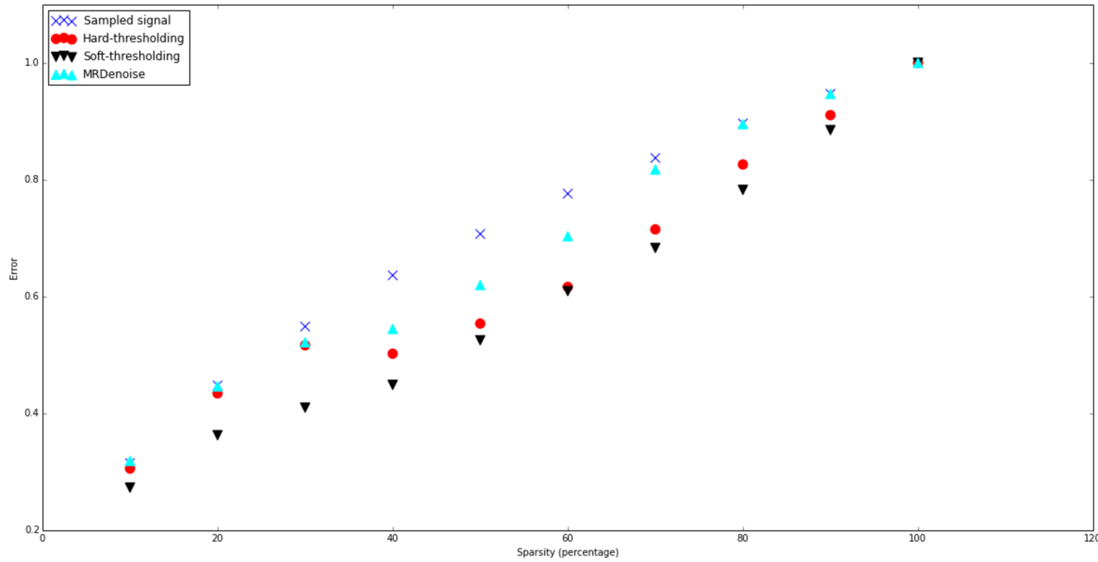


Figure 6: Error as a function of the sparsity

We can see that the algorithms seem to work for reconstructing images. The error is significantly lower than the sampled signal for each of the four algorithms as long as the sparsity is not 100%. The soft-thresholding method seem to work best : it has the lowest error for each of the 10 experiments (a range of sparsity percentages from 10% to 100%).

In the appendix, we provide qualitative results on a denoising task on 3 different images, using the 3 algorithms. The results are found in Figure 7.

## 5 Conclusion

In this report, we presented the starlet transform which allows to reconstruct and denoise astronomical images, through the use of 3 algorithms : Hard, soft-thresholding and Iterative multiresolution mask denoising algorithm. The results depend on the quantity of noise and/or destruction of the image. The algorithms mostly succeed in denoising and reconstructing the images. It is extremely interesting to see how compressed sensing is relevant to the Astronomy field. Creating a mathematically well-founded framework in Statistics such as compressed sensing is one thing, creating a useful framework for other scientists is another, and some might argue that the latter one is more important. As statisticians, we should listen to the needs of other scientists and try to focus on research that aims to help them.

## Bibliography


[1] T. Blumensath and M. E. Davies. Iterative hard thresholding for compressed sensing. *CoRR*, abs/0805.0510, 2008. URL `http://arxiv.org/abs/0805.0510`.

[2] J. Bobin, J.-L. Starck, and R. Ottensamer. Compressed sensing in astronomy. *IEEE Journal of Selected Topics in Signal Processing*, 2(5):718–726, 2008.

[3] K. Bredies and D. A. Lorenz. Linear convergence of iterative soft-thresholding. *Journal of Fourier Analysis and Applications*, 14(5-6):813–837, 2008.

[4] J. Starck, F. Murtagh, and A. Bijaoui. Image restoration with denoising using multi-resolution. In *The Restoration of HST Images and Spectra-II*, page 111, 1994.

[5] J.-L. Starck, F. Murtagh, and J. M. Fadili. *Sparse image and signal processing: wavelets, curvelets, morphological diversity*. Cambridge university press, 2010.

## 6 Appendix

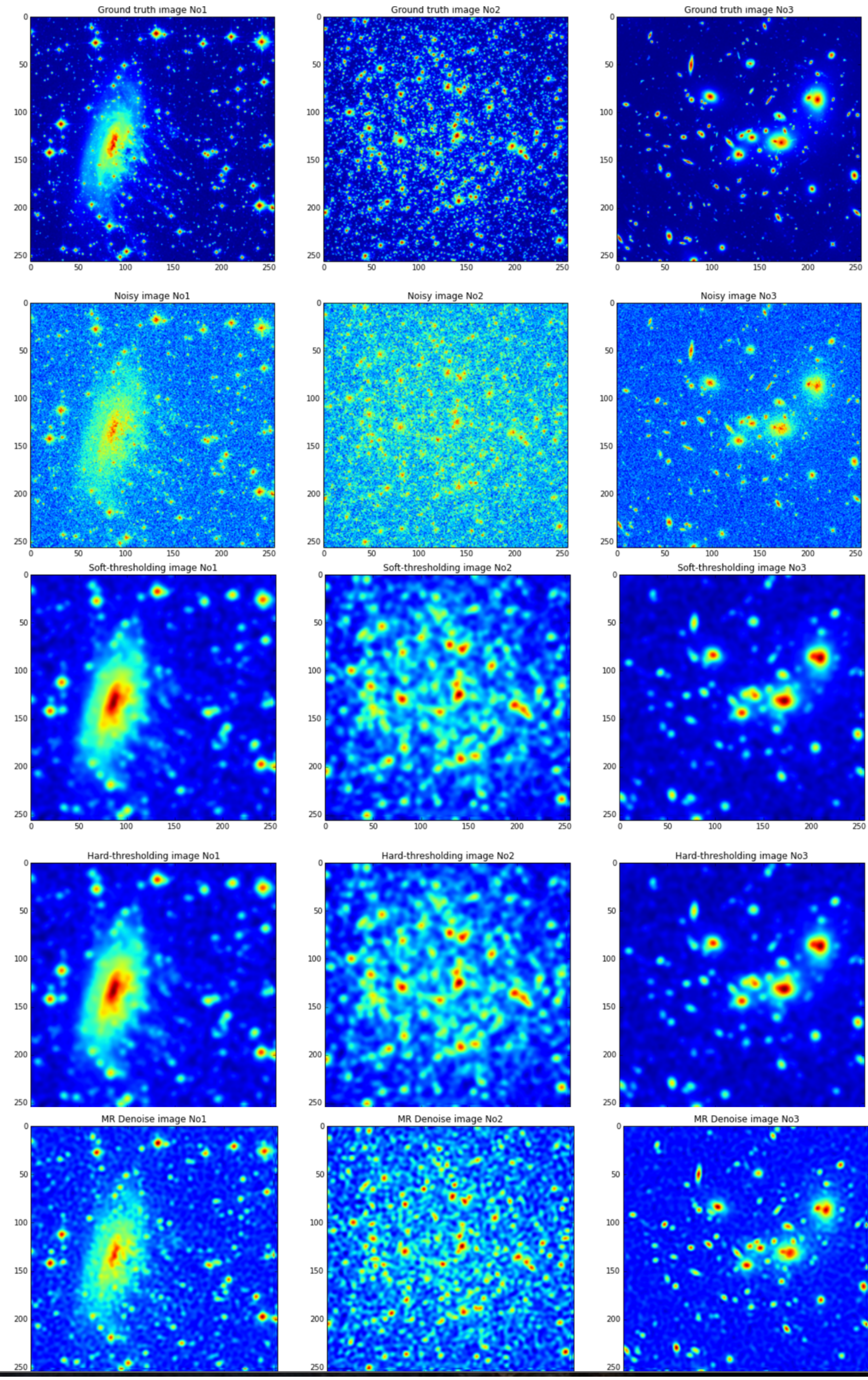


Figure 7: Example on 3 new images