**Point 1:** The novelty of the paper is limited, considering nuclear norm penalty, soft thresholding, and D-Amp are widely used. Many important references in the field are missing.

**Response 1:** We have done our best to show our idea:
(a) modeling the signal with the LSM Prior to describe the group sparsity of residual;
(b) solving the model with an entropy-based algorithm (i.e., EM method) because of the latent variables;
(c) the M step of the EM method corresponds to the D-AMP algorithm;
(d) the residual learning is a step of the D-AMP algorithm.
We believe that our work is somewhat innovative.

References in our manuscript about nuclear norm penalty can be found in [29,35,36]
References in our manuscript about soft thresholding can be found in [5,40,41,42]
References in our manuscript about D-Amp can be found in [30,31]

According to your advice, we add the following reference which is about nuclear norm penalty:

**Point 2:** The revised paper failed to provide a detailed comparison with regards to the steps, penalties, complexity, convergence analysis to other methods.

**Response 2:** Several modifications have been made in page 10 and marked by yellow.

The modifications are as follows:

“We can compare three D-AMP related algorithms: RL-DAMP, BM3D-AMP and LR-AMP in terms of steps, penalties, complexity, convergence, and then validate them in our following experiments. Three D-AMP related algorithms all have two steps: a denoising and a residual update step. The sparse penalties of RL-DAMP and LR-AMP can be showed in Eq. (7) and Eq. (23) respectively, while BM3D-AMP do not have explicit penalties, which uses the BM3D denoiser as the implicit sparse terms. Note that, these sparse penalties pay a role of the denoising step in the general steps of the D-AMP related algorithms. Recall that, G is the total number of similar patch groups, the size of the low-rank matrices is $M \times N$, where M is the number of similar patches, and the size of an image patch is $\sqrt{N} \times \sqrt{N}$. The most time-consuming operation in our method is that performing the SVD on the low-rank matrix to get the dictionary. The complexity of this step is $O(GM^3)$. This step is also the most time-consuming operation LR-AMP, thus, its complexity also $O(GM^3)$. The most time-consuming operation of BM3D-AMP is that performing the 3D wavelet transform on the 3D image cube. The complexity of this step is $O(G\sqrt{N})$. Note that $M \geq N$, thus, the complexity of RL-DAMP and LR-AMP is higher than the one of BM3D-AMP. As the BM3D method is written in C language, the running time of the BM3D-AMP method is much shorter. These three methods are based on the D-AMP algorithm, and inherit its convergence. The convergence can be validated partly in the following experiments with the iteration number vs. PSNR curves and CPU time vs. PSNR curves.”
**Point 3:** The comparison algorithms are using the default parameters, which are far from what they can really achieve.

**Response 3:** In our experiments, we use the default parameters for the reason that parameter optimization of RL-DAMP, BM3D-AMP, LR-AMP and ADMM-Net have been done by the authors of the original paper. Only the NLR-CS method which was not designed for this kind of sampling matrix, causing a bit of poor performance. However, the authors of NLR-CS claim that their experiments contain this kind of sampling matrix, but they do not provide the results in the paper (they do not provide this kind of sampling matrix in their code). Thus, we may think that it is considered a good choice not to change the parameters.