

Unsupervised Adaptation Learning for Hyperspectral Imagery Super-resolution

Lei Zhang^{1*}, Jiangtao Nie^{2*}, Wei Wei^{2,3}, Yanning Zhang², Shengcai Liao¹ and Ling Shao¹

¹Inception Institute of Artificial Intelligence (IIAI), Abu Dhabi, United Arab Emirates

²School of Computer Science, Northwestern Polytechnical University, Xi'an, China

³Northwestern Polytechnical University in Shenzhen, Shenzhen, China

Abstract

The key for fusion based hyperspectral image (HSI) super-resolution (SR) is to infer the posteriori of a latent HSI using appropriate image prior and likelihood that depends on degeneration. However, in practice the priors of high-dimensional HSIs can be extremely complicated and the degeneration is often unknown. Consequently most existing approaches that assume a shallow hand-crafted image prior and a pre-defined degeneration, fail to well generalize in real applications. To tackle this problem, we present an unsupervised adaptation learning (UAL) framework. Instead of directly modelling the complicated image prior, we propose to first implicitly learn a general image prior using deep networks and then adapt it to a specific HSI. Following this idea, we develop a two-stage SR network that leverages two consecutive modules: a fusion module and an adaptation module, to recover the latent HSI in a coarse-to-fine scheme. The fusion module is pretrained in a supervised manner on synthetic data to capture a spatial-spectral prior that is general across most HSIs. To adapt the learned general prior to the specific HSI under unknown degeneration, we introduce a simple degeneration network to assist learning both the adaptation module and the degeneration in an unsupervised way. In this way, the resultant image-specific prior and the estimated degeneration can benefit the inference of a more accurate posteriori, thereby increasing generalization capacity. To verify the efficacy of UAL, we extensively evaluate it on four benchmark datasets and report strong results that surpass existing approaches.

1. Introduction

Hyperspectral images (HSIs) consists of hundreds of spectral bands that record the reflectance of an imaging scene across a consecutive wavelengths with narrow interval (e.g. 10nm) [1, 32], where each pixel contains a spec-

*The first two authors contributed equally. The corresponding author is Wei Wei (email: weiweinwpu@nwpu.edu.cn). Project page: <https://github.com/JiangtaoNie/UAL>

trum. Due to the discriminative power of their spectra [13], HSIs have been widely applied to various computer vision tasks, such as target detection [17], classification [13], tracking [22] etc. However, physical limitations on spectral sensor [12, 7] often prevent the collection of high resolution (HR) HSIs in practice. Hence increasing efforts have been put forwards to HSI super-resolution (SR).

Recent studies [4, 14, 19] have shown that fusing a low resolution (LR) HSI with an HR multispectral image (MSI) using maximum a posteriori (MAP) estimation [32] is a promising approach for HSI SR. The key lies on inferring the posteriori of the latent HSI using an appropriate image prior and the likelihood determined by the degeneration from the latent HSI to the observed LR HSI. To this end, existing approaches have handcrafted various shallow prior models, e.g. sparsity [12, 8], low-rank [30] etc. Together with pre-defined degeneration models, e.g. Gaussian blur based down-sampling [8, 30], they can successfully recover simple structures of HSIs under given degenerations. However, due to the countless number of imaging scenes that exist, real HSIs often contain abundant complex structures and the true prior can be extremely complicated, e.g. multi-mode, image-specific. In addition, random factors in imaging procedure (e.g. sensor shaking, noise corruption) usually introduce unknown degeneration into observations. Hence, when applied to real scenarios, most existing approaches fail to generalize appropriately or obtain pleasing SR performance, as shown in Figure 1.

In this study, we propose an unsupervised adaptation learning (UAL) framework and demonstrate good generalization efficacy in HSI SR. In contrast to existing approaches, we propose to learn a deep image-specific prior for each HSI. Instead of directly modelling such a complicated prior, we first utilize deep networks to implicitly learn an image prior that is general across most HSIs and then adapt it to a specific HSI. Following this idea, we propose a two-stage SR network that employs two consecutive modules to recover the latent HSI in a coarse-to-fine scheme: a *fusion module* and an *adaptation module*. The fusion module adopts a brand-new mutual-guiding architecture, and is pretrained

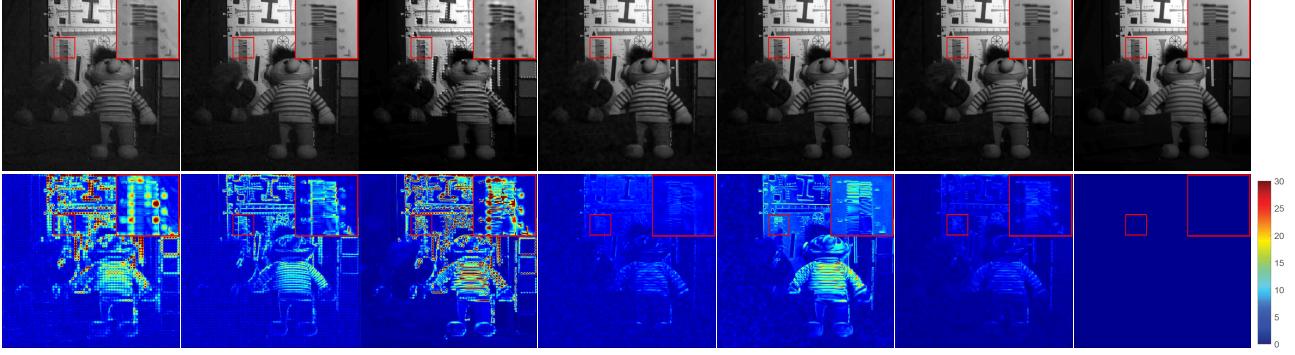


Figure 1. Visual SR results and reconstruction error maps on an image from the CAVE dataset [28] with unknown degeneration (e.g., no-Gaussian blur and random noise) when SR scale s is 8. Most existing methods shows obvious artifacts and reconstruction error.

on synthetic data in a supervised manner to learn a general spatial-spectral prior, which it employs for rough SR recovery. To adapt the learned general image prior to a specific HSI, the key is to extract image-specific knowledge from two observed images under unknown degeneration. To this end, we introduce a simple degeneration network to assist learning both the adaptation module and the degeneration in an unsupervised manner. By doing this, we can jointly obtain an image-specific prior and estimate the degeneration, ultimately inferring the posteriori of the latent HSI accurately and increasing the generalization capacity. Extensive experiments demonstrate the efficacy of UAL in generalizing to unknown degenerations and different datasets.

In summary, we provide four main contributions: 1) we present an UAL framework to generalize fusion based SR to real cases; 2) we propose to learn the complicated image prior via adapting deep general image priors to a specific image; 3) we develop a new mutual-guiding module for images fusion, which is an universal module and can be plugged into any other fusion or multi-input networks; 4) we show state-of-the-art SR performance on four datasets.

2. Related works

According to the image prior utilized, existing fusion-based SR methods can be divided into two categories.

Shallow image prior Existing methods have handcrafted various shallow image prior models [24, 3, 12, 8, 30]. For example, Wycoff *et al.* [24] and Lanaras [12] propose to decompose the latent HR HSI into a non-negative end-member matrix and an abundance matrix, and then impose a sparse prior on the abundance matrix. Akhtar [2] exploit the signal sparsity, nonnegativity and spatial structure of the latent HSI by projecting each spectrum onto a pre-extracted spectral dictionary. In [3], a Bayesian sparse representation scheme is utilized to infer the probability distributions of spectra and their proportions decomposed from the latent HSI. Recently, Dong *et al.* [8] further consider the spatially non-local similarity of the latent HSI. Zhang *et al.* [30]

exploit the spatial manifold structure. Due to the limited expressiveness of shallow image priors, these methods fail to generalize well in challenging cases, especially when the degeneration in observation is unknown [15, 10]. In contrast, we learn a deep image-specific prior for each HSI. The deep structure and image-specific nature provide good generalization performance in practice.

Deep image prior Inspired by the great success of deep learning, several deep image priors have been proposed for HSIs. For example, Ulyanov *et al.* [21] employ a randomly initialized deep convolutional neural network as an image prior and achieve unsupervised image recovery under a given degeneration. When the degeneration is unknown, further introducing a degeneration network seems to be helpful [5]. However, since the observation only contains limited information for the latent image, unsupervised learning in this case often results in over-fitting and incurs expensive computational costs. With special design, the proposed UAL effectively avoids over-fitting and reduces the computational cost. Very recently, Xie et al [25] implicitly learn a deep image prior by training a SR network under supervision. The learned prior can effectively capture the general structure, however fails to depict the image-specific details and cope with unknown degenerations. In contrast, the proposed UAL can address these problems successfully.

3. Unsupervised adaptation learning

3.1. Problem formulation

Given two observed images, including an HR MSI $\mathbf{X} \in \mathbb{R}^{b \times N}$ consisting of b spectral bands with N pixels in the spatial domain and an LR HSI $\mathbf{Y} \in \mathbb{R}^{B \times n}$, the fusion-based HSI SR aims to produce a latent HR HSI $\mathbf{Z} \in \mathbb{R}^{B \times N}$ (e.g. $b \ll B, n \ll N$). The relation between the latent \mathbf{Z} and the other two observations can be formulated as

$$\mathbf{X} = \mathbf{P}\mathbf{Z}, \quad \mathbf{Y} = \mathbf{Z}\mathbf{H}, \quad (1)$$

where $\mathbf{P} \in \mathbb{R}^{b \times B}$ denotes the spectral response function, and $\mathbf{H} \in \mathbb{R}^{N \times n}$ represents the degeneration from \mathbf{Z} to \mathbf{Y} .

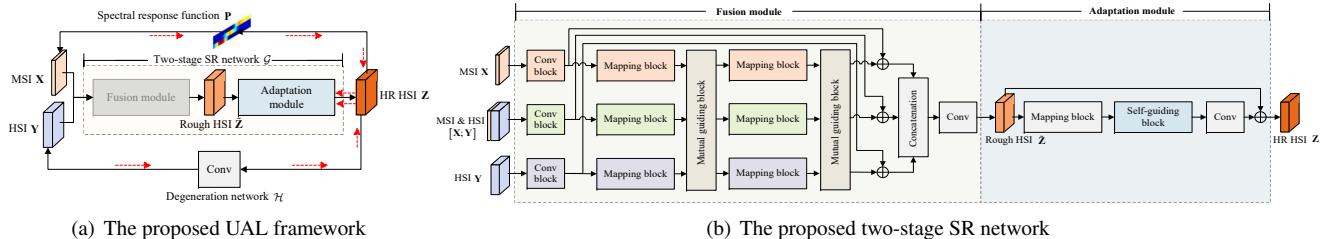


Figure 2. Flow chart of the proposed UAL and the proposed two-stage SR network. In figure (a), red arrows indicate the back-propagated gradient in unsupervised learning, and we denote the fusion module in gray to indicate that its weights are fixed in unsupervised learning.

When both \mathbf{P} and \mathbf{H} are predefined, the MAP estimation of fusion-based HSI SR can be formulated as

$$\min_{\mathbf{Z}} \|\mathbf{X} - \mathbf{P}\mathbf{Z}\|^2 + \|\mathbf{Y} - \mathbf{Z}\mathbf{H}\|^2 + \lambda\mathcal{R}(\mathbf{Z}), \quad (2)$$

where the first two terms represent the likelihood-based energy and $\mathcal{R}(\mathbf{Z})$ denotes the regularization introduced by image prior on \mathbf{Z} . Since \mathbf{P} is determined by the parameters of camera and often can be obtained in advance [12, 8, 14, 30], in this study we assume \mathbf{P} is given and mainly focus on coping with the unknown degeneration \mathbf{H} in practice.

3.2. Unsupervised adaptation learning

Instead of separately modelling the likelihood and prior as in Eq. (2), we employ a deep SR network to directly model the maximum posterior of \mathbf{Z} , which learns the image prior implicitly during network training. To learn an image-specific prior under unknown degeneration, we first design a two-stage SR network as shown in Figure 2(a), which consists of two modules: a fusion module and an adaptation module. Then, we carry out the following two steps to accomplish prior learning and degeneration estimation.

Pre-train fusion module We train the fusion module in a supervised manner to learn a general prior and utilize it for rough SR recovery. To this end, we first employ extensive synthetic degenerations $\{\mathbf{H}_j\}$ to produce training pairs on a set of HSIs $\{\mathbf{Z}_i\}$ (see details in Section 4.1). Then, we solve the following problem to learn the fusion module.

$$\begin{aligned} & \min_{\theta_f} \sum_{i,j} \|\mathbf{Z}_i - \hat{\mathbf{Z}}_{ij}\|^2, \\ & \text{s.t., } \hat{\mathbf{Z}}_{ij} = \mathcal{F}(\mathbf{X}_{ij}, \mathbf{Y}_{ij}; \theta_f), \mathbf{X}_{ij} = \mathbf{P}\mathbf{Z}_i, \mathbf{Y}_{ij} = \mathbf{Z}_i\mathbf{H}_j, \end{aligned} \quad (3)$$

where $\{\mathbf{Z}_i, \mathbf{X}_{ij}, \mathbf{Y}_{ij}\}$ denotes the training pair generated by the degeneration \mathbf{H}_j . \mathcal{F} denotes the fusion module parameterized by θ_f . $\|\cdot\|$ denotes the reconstruction loss.

Unsupervised adaptation learning To adapt the general image prior learned in the fusion module to a specific HSI under unknown degeneration, the solution lies on exploiting the image-specific statistics. Previous work [33] has shown that the degenerated observation often contains some image-specific statistics, and can supervise the learning of an image-specific prior when the degeneration is given [21].

To generalize this idea to real cases with unknown degenerations, we introduce a degeneration network together with the given spectral response function \mathbf{P} to map the image \mathbf{Z} generated from the fusion module back into the observed \mathbf{X} and \mathbf{Y} , respectively, as shown in Figure 2(a). In this way, both the adaptation module and degeneration can be jointly learned in an unsupervised manner as

$$\begin{aligned} & \min_{\theta, \vartheta} \|\mathbf{X} - \mathbf{P}\mathbf{Z}\|^2 + \|\mathbf{Y} - \mathbf{Z}\mathbf{H}\|^2, \\ & \text{s.t., } \mathbf{Z} = \mathcal{G}(\mathbf{X}, \mathbf{Y}; \theta), \mathbf{Z}\mathbf{H} = \mathcal{H}(\mathbf{Z}; \vartheta), \end{aligned} \quad (4)$$

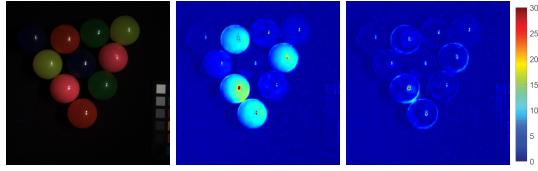
where $\mathcal{G}(\cdot; \theta)$ denotes the proposed two-stage SR network that takes both observed images as input and outputs the latent HSI. In Eq. (4), we fix the fusion module, and let θ denote the parameters in the adaptation module. $\mathcal{H}(\cdot; \vartheta)$ denotes the degeneration network parameterized by ϑ .

The UAL proposed above provides a promising way to jointly achieve degeneration estimation and HSI SR with a deep image-specific prior. However, due to the limited amount of data in \mathbf{X} and \mathbf{Y} , designing casual architectures for the SR and degeneration networks may cause over-fitting and incur expensive computational costs in unsupervised learning. To address this problem, we propose to largely reduce the number of parameters to learn in the unsupervised learning via two strategies: 1) we implement the degeneration network \mathcal{H} with a very simple architecture; 2) we construct the fusion module with a complex architecture so that it contains most parameters of the SR network, while adopting a light-weight architecture with only a few parameters as the adaptation module. By doing these, only a few parameters will be learned in Eq. (4), thus effectively avoiding over-fitting and reducing computational cost. In the following, we will discuss these two strategies in detail.

3.3. Simple degeneration network

Before designing the degeneration network \mathcal{H} , we first revisit the physical structure of the degeneration \mathbf{H} . In practice, \mathbf{H} involves both spatial down-sampling and noise corruption [9]. Down-sampling often convolves the image with a specific kernel beforehand [9, 29], while noise corruption is close to the additive zero-mean random noise [18, 31]. Thus, degeneration \mathbf{H} on \mathbf{Z} can be formulated as

$$\mathbf{Z}\mathbf{H} = (\mathbf{Z} \circledast \mathbf{k}) \downarrow_s + \mathbf{N}, \quad (5)$$



(a) Ground truth (b) Fusion module (c) SR network

Figure 3. Reconstruction error maps generated by the pre-trained fusion module and the proposed two-stage SR network.

where \mathbf{k} denotes a 2-D spatial kernel and \circledast indicates convolving each band of \mathbf{Z} with the same kernel \mathbf{k} . \downarrow_s represents the down-sampling operation in the spatial domain with a scaling factor s . \mathbf{N} denotes the random noise corruption. As shown in Eq (5), the major parameter in \mathbf{H} is the kernel \mathbf{k} , e.g. a $k \times k$ -sized \mathbf{k} that only involves k^2 parameters with $k < 30$. Inspired by this, we adopt a single convolutional layer with a kernel \mathbf{k} and a stride s as the degeneration network \mathcal{H} . Since noise corruption \mathbf{N} can be absorbed into the loss function in Eq. (4) as discussed in [21], no specific structure is further designed for the noise \mathbf{N} .

3.4. Two-stage SR network

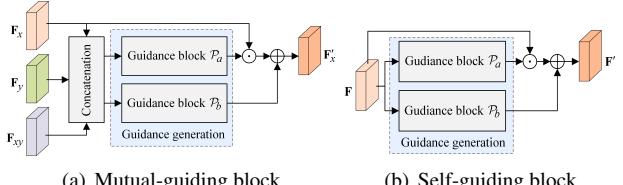
As shown in Figure 2(b), we implement the fusion module with a new mutual-guiding architecture to exploit the general spatial-spectral prior for rough SR recovery, while implementing the adaptation module with a light-weight architecture to refine image-specific details. This enables a coarse-to-fine SR, as shown in Figure 3. In the following, we discuss both modules in detail.

3.4.1 Mutual-guiding fusion module

For deep fusion-based HSI SR, a direct way is to concatenate the observed images \mathbf{X} and \mathbf{Y} along the spectral dimension and feed them into an appropriate deep SR network. The concatenation enables the correlation between two observed images to be exploited. However, it fails to explicitly leverage specific knowledge from each input.

To sufficiently utilize both intra-input and inter-input knowledge, we propose a mutual-guiding fusion module that takes three different inputs, including the HR MSI \mathbf{X} , LR HSI \mathbf{Y} ¹ and their concatenation $[\mathbf{X}; \mathbf{Y}]$. As shown in Figure 2(b), this module first utilizes three parallel branches to separately map the three inputs into deep features, and then the concatenation of deep features is convolved to produce a rough estimate $\tilde{\mathbf{Z}}$ for the latent HSI \mathbf{Z} . Each branch consists of two basic blocks, including a mapping block and a mutual-guiding block. The mapping block stacks M convolutional blocks (*i.e.*, a convolutional layer followed by a ReLu [16]), while the mutual guiding block separately transforms each feature from the three branches under the guidance of the other two features. Specifically, let \mathbf{F}_x ,

¹In this study we employ bicubic interpolation to upsample the spatial dimension of \mathbf{Y} to the same as that of \mathbf{X} as input.



(a) Mutual-guiding block (b) Self-guiding block

Figure 4. Architecture of mutual-guiding and self-guiding blocks. \mathbf{F}_y and \mathbf{F}_{xy} denote three features produced by a mapping block. Taking \mathbf{F}_x as an example, the mutual guiding block linearly transforms \mathbf{F}_x as

$$\mathbf{F}'_x = \mathbf{F}_x \odot \mathbf{A} + \mathbf{B}, \quad (6)$$

where \mathbf{F}'_x denotes the transformed \mathbf{F}_x , and \odot indicates point-wise product. \mathbf{A} and \mathbf{B} represent the guidance. Similar to [23], this transformation regulates each entry of \mathbf{F}_x accordingly, thus improving the flexibility in fusion. We compute \mathbf{A} and \mathbf{B} separately using two individual guidance blocks as shown in Figure 4(a), which can be formulated as

$$\mathbf{A} = \mathcal{P}_a([\mathbf{F}_x; \mathbf{F}_y; \mathbf{F}_{xy}]; \pi_a); \mathbf{B} = \mathcal{P}_b([\mathbf{F}_x; \mathbf{F}_y; \mathbf{F}_{xy}]; \pi_b), \quad (7)$$

where $\mathcal{P}_a(\cdot; \pi_a)$ and $\mathcal{P}_b(\cdot; \pi_b)$ denote the guidance blocks for \mathbf{A} and \mathbf{B} , respectively. $[\mathbf{F}_x; \mathbf{F}_y; \mathbf{F}_{xy}]$ denotes the concatenation of three features. In this study, we stack M convolutional blocks as \mathcal{P}_a and \mathcal{P}_b . To avoid the abuse of notations, we utilize the same \mathcal{P}_a and \mathcal{P}_b to separately transform \mathbf{F}_x , \mathbf{F}_y and \mathbf{F}_{xy} as Eq. (6). When computing the guidance for \mathbf{F}_y and \mathbf{F}_{xy} , we employ the concatenation $[\mathbf{F}_y; \mathbf{F}_x; \mathbf{F}_{xy}]$ and $[\mathbf{F}_{xy}; \mathbf{F}_x; \mathbf{F}_y]$, respectively.

3.4.2 Self-guiding adaptation module

To recover image-specific details, we implement the adaptation module using a residual architecture, as shown in Figure 2(b), where the backbone recovers the residual between the HSI $\tilde{\mathbf{Z}}$ estimated by the fusion module and the ground truth HSI \mathbf{Z} . Let \mathcal{A} denote the backbone, the output $\hat{\mathbf{Z}}$ of the adaptation module can be given as

$$\hat{\mathbf{Z}} = \tilde{\mathbf{Z}} + \mathcal{A}(\tilde{\mathbf{Z}}). \quad (8)$$

As shown in Figure 2(b), the backbone consists of three components, including a mapping block, which is same as that in fusion module, a self-guiding block and a convolutional layer for output. The self-guiding block has a similar architecture to that of the mutual-guiding block in Section 3.4.1, shown as Figure 4(b). The only difference is that only the input feature itself is utilized to compute the guidance. Let \mathbf{F} denote the feature fed into this block, the output can be obtained from Eq. (6) with the following guidance

$$\mathbf{A} = \mathcal{P}_a(\mathbf{F}; \pi_a); \mathbf{B} = \mathcal{P}_b(\mathbf{F}; \pi_b). \quad (9)$$

\mathcal{P}_a and \mathcal{P}_b again denote the guidance blocks. In unsupervised learning, this block learns to adaptively adjust the pixel-wise residual and refine image-specific details, as shown in Figure 3.

4. Experiments

4.1. Experimental settings

Datasets We adopt three benchmark HSI datasets and one real HSI dataset for evaluation, including CAVE [28], Harvard [6], NTIRE [20] and HypSen [27]. In the CAVE dataset, there are 32 HSIs each of which has 512×512 pixels and 31 spectral bands in a wavelength range of 400nm to 700nm. Harvard dataset consists of 50 HSIs. Each image contains 1392×1040 pixels and 31 spectral bands in a wavelength range of 420nm to 720nm. In the NTIRE dataset, there are 255 HSIs each of which is of size 1392×1300 in spatial domain and contains 31 bands in a wavelength range of 400nm to 700nm. In contrast to these three datasets, HypSen [27] is a real fusion-based HSI dataset. It consists of an 30m-resolution HSI captured by Hyperion sensor on the Earth Observing-1 satellite, and an 10m-resolution MSI produced by Sentinel-2A satellite. After removing noisy and water absorption bands as [27], the LR HSI consists of 84 spectral bands, while the HR MSI contains 13 spectral bands. In our experiment, we crop a subimage of size 250×330 from the HSI and one of size 750×990 from the MSI, and make sure both capture the same scene.

Comparison methods & Evaluation metrics For comparison, we select five state-of-the-art fusion based HSI SR methods, including NSSR [8], CMS [30], MHF-net [25], YONG [15] and DIP [21]. Except YONG [15], all other methods predefine a degeneration $\hat{\mathbf{H}}$ as Eq. (5) using a Gaussian kernel but no noise corruption. Among them, NSSR [8] and CMS [30] exploit the sparse or manifold prior of the latent HSI. MHF-net [25] train a deep network in a supervised manner to recover the latent HSI, while DIP [21] employs a deep image prior for SR in an unsupervised way. YONG [15] cast HSI SR and degeneration estimation into a joint learning framework but with a shallow image prior.

To quantitatively evaluate the SR performance, we adopt four standard metrics: root-mean-square error (RMSE)², peak signal-to-noise ratio (PSNR), spectral angle mapper (SAM) and structural similarity index (SSIM).

Implementation details In our experiment, we consider each HSI in the benchmark datasets as the ground truth of the latent HSI \mathbf{Z} . Given \mathbf{Z} , we utilize the spectral response function of a Nikon D700 camera in [8, 30] as \mathbf{P} to produce an HR MSI \mathbf{X} as Eq. (1). To obtain the observed LR HSI \mathbf{Y} , we first down-sample \mathbf{Z} with a kernel \mathbf{k} and scale s as Eq. (5), and then add Gaussian white noise into the down-sampled result. To simulate various degenerations in practice, we separately employ four non-Gaussian kernels, as shown in Figure 5, as well as three different levels of noise corruption, *e.g.* 30db, 35db, 40db (the signal-to-noise ratio of \mathbf{Y}), to generate \mathbf{Y} . Given \mathbf{X} , \mathbf{Y} and \mathbf{P} , we employ all methods to recover the latent \mathbf{Z} . To show their performance

²Following [8, 30], we also compute the RMSE on the 8-bit image.

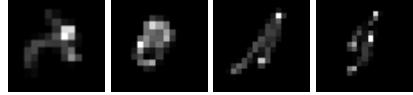


Figure 5. Four different kernels utilized to produce the degeneration \mathbf{H} in test. From left to right, they are \mathbf{K}_1 , \mathbf{K}_2 , \mathbf{K}_3 and \mathbf{K}_4 .

under unknown degeneration, we define another degeneration $\hat{\mathbf{H}}$ using a 8×8 sized Gaussian kernel with standard deviation $\sqrt{3}$ as [8, 30], and incorporate it into baselines without degeneration estimation as the pre-defined degeneration. Fair comparison, MHF-net [25] is trained on the same training set as the proposed UAL.

In the proposed SR network, we set $M=3$ and equip each convolution layer with 64 kernels of size 3×3 . In the degeneration network \mathcal{H} , we fix the kernel size as 32×32 . To pre-train the fusion module, we respectively select 12 HSIs from the CAVE, 20 from the Harvard and 20 from NTIRE as training sets. While the remaining 20 HSIs in CAVE, 30 in Harvard and the other 50 in NTIRE are utilized in test. Given a training HSI, we extract patches of size 128×128 with stride 64 to generate training pairs as above but this time using synthetic kernels based degeneration, *i.e.*, Gaussian kernels with spatial size and standard deviation randomly sampled in the ranges [5, 15] and [0.5, 2], respectively. Noted that no kernel in test is utilized for training. In this study, we implement UAL in Pytorch. To train the fusion module, we adopt the Adam optimizer [11] with a ℓ_1 norm based loss. We initialize the learning rate as 1e-4 and decay it every 10 epochs by 0.7. The batch size is set as 6, and the training procedure is terminated in 150 epochs. In the test phase, we feed two observed images \mathbf{X} and \mathbf{Y} into the SR network and then train the adaptation module and the degeneration network with a ℓ_1 norm based loss. For optimization, we utilize the Adam optimizer with initial learning rates 9e-5, 1e-4 and weight decay 1e-3, 1e-5 to learn the adaptation module and the degeneration network, respectively. We terminate the learning process in 1500 epochs.

4.2. Ablation studies

In this section, we conduct ablation study on the CAVE dataset. Without loss of generality, we produce the test degeneration with the kernel \mathbf{k}_1 in Figure 5 and 40db Gaussian white noise. The SR scale is fixed as 8.

Effect of components in UAL The proposed UAL has three key components: 1) a two-stage SR network; 2) a pre-trained fusion module whose weights are fixed in unsupervised learning; 3) a degeneration estimation module. To show the effect of the two-stage SR network, we utilize two variants of UAL that remove the adaptation module and the fusion module, respectively. They are termed ‘UAL w/o adaptation’ and ‘UAL w/o fusion’. Their results are given in Table 1. As can be seen, removing either module causes obvious performance drop, and the performance drops more when removing the fusion module, *e.g.* PSNR drops

Table 3. Performance under degeneration produced by different levels of noise corruption and kernel k_1 with SR scale $s=8$.

| Methods | 30db | | | | 35db | | | | 40db | | | |
|--------------|-------------|--------------|--------------|---------------|-------------|--------------|-------------|---------------|-------------|--------------|-------------|---------------|
| | RMSE | PSNR | SAM | SSIM | RMSE | PSNR | SAM | SSIM | RMSE | PSNR | SAM | SSIM |
| NSSR [8] | 11.45 | 27.17 | 34.64 | 0.5661 | 9.31 | 29.17 | 25.76 | 0.7477 | 8.22 | 32.10 | 16.58 | 0.9229 |
| CMS [30] | 7.96 | 30.24 | 31.88 | 0.6596 | 5.47 | 33.58 | 23.48 | 0.8118 | 4.28 | 36.52 | 14.88 | 0.9517 |
| DIP [21] | 11.95 | 27.68 | 21.52 | 0.8298 | 11.54 | 27.96 | 14.93 | 0.8782 | 9.56 | 29.67 | 15.05 | 0.9117 |
| MHF-net [25] | 4.55 | 35.06 | 24.68 | 0.8146 | 3.24 | 38.17 | 17.88 | 0.9065 | 2.62 | 40.26 | 12.90 | 0.9528 |
| YONG [15] | 3.64 | 37.16 | 13.95 | 0.9351 | 3.32 | 38.07 | 11.48 | 0.9533 | 3.12 | 38.73 | 9.63 | 0.9646 |
| UAL | 2.80 | 39.43 | 13.46 | 0.9566 | 2.23 | 41.59 | 9.36 | 0.9777 | 1.85 | 43.23 | 6.72 | 0.9862 |

Table 4. Performance when SR scale s is 8 and the degeneration is produced by k_1 and 40db noise.

| Methods | CAVE | | | | Harvard | | | | NTIRE | | | |
|------------------------|-------------|--------------|-------------|---------------|-------------|--------------|-------------|---------------|-------------|--------------|-------------|---------------|
| | RMSE | PSNR | SAM | SSIM | RMSE | PSNR | SAM | SSIM | RMSE | PSNR | SAM | SSIM |
| NSSR [8] | 8.22 | 32.10 | 16.58 | 0.9229 | 4.18 | 36.28 | 3.82 | 0.9632 | 3.28 | 38.56 | 2.18 | 0.9792 |
| CMS [30] | 4.28 | 36.52 | 14.88 | 0.9517 | 2.71 | 39.89 | 3.88 | 0.9683 | 2.01 | 42.63 | 1.68 | 0.9913 |
| DIP [21] | 9.56 | 29.67 | 15.05 | 0.9117 | 10.51 | 28.33 | 8.08 | 0.8788 | 8.17 | 31.15 | 5.05 | 0.9085 |
| MHF-net (CAVE) [25] | 2.62 | 40.26 | 12.90 | 0.9528 | 3.23 | 38.12 | 6.19 | 0.9460 | 2.59 | 39.92 | 2.77 | 0.9851 |
| MHF-net (Harvard) [25] | 7.79 | 30.93 | 25.30 | 0.8539 | 2.29 | 41.11 | 4.21 | 0.9714 | 8.55 | 30.59 | 11.56 | 0.9227 |
| MHF-net (NTIRE) [25] | 5.68 | 33.94 | 16.02 | 0.9262 | 3.38 | 38.03 | 5.03 | 0.9639 | 1.70 | 44.15 | 1.90 | 0.9934 |
| YONG [15] | 3.12 | 38.73 | 9.63 | 0.9646 | 2.57 | 40.43 | 4.08 | 0.9690 | 4.92 | 34.95 | 3.01 | 0.9685 |
| UAL (CAVE) | 1.85 | 43.23 | 6.72 | 0.9862 | 2.36 | 41.19 | 3.39 | 0.9725 | 1.21 | 46.09 | 1.31 | 0.9951 |
| UAL (Harvard) | 2.55 | 40.51 | 7.65 | 0.9790 | 2.08 | 42.38 | 2.67 | 0.9815 | 1.78 | 44.49 | 1.14 | 0.9953 |
| UAL (NTIRE) | 2.75 | 39.78 | 8.84 | 0.9707 | 2.41 | 41.07 | 3.13 | 0.9753 | 1.50 | 46.19 | 0.99 | 0.9963 |

Table 1. Effect of each component in the proposed UAL.

| Methods | RMSE | PSNR | SAM | SSIM |
|----------------------------|-------------|--------------|-------------|---------------|
| UAL w/o adaptation | 2.35 | 41.66 | 6.87 | 0.9833 |
| UAL w/o fusion | 4.18 | 36.17 | 10.63 | 0.9484 |
| UAL w/o pre-train | 3.50 | 37.51 | 6.50 | 0.9735 |
| UAL (fine-tune) | 2.54 | 40.27 | 5.71 | 0.9843 |
| UAL ($\hat{\mathbf{H}}$) | 4.81 | 38.58 | 8.43 | 0.9592 |
| UAL (only concat) | 3.73 | 37.42 | 8.35 | 0.9653 |
| UAL w/o concat | 2.05 | 42.46 | 6.74 | 0.9846 |
| UAL w/o mutual-guiding | 2.78 | 39.98 | 7.44 | 0.9782 |
| UAL w/o residual | 3.45 | 37.77 | 6.68 | 0.9736 |
| UAL w/o self-guiding | 1.95 | 42.94 | 6.74 | 0.9860 |
| UAL | 1.85 | 43.23 | 6.72 | 0.9862 |

Table 2. Performance under degeneration produced by different kernels and 40db noise with SR scale $s=8$.

| k_i | Methods | RMSE | PSNR | SAM | SSIM |
|-------|--------------|-------------|--------------|-------------|---------------|
| | | 8.22 | 32.10 | 16.58 | 0.9229 |
| k_1 | NSSR [8] | 8.22 | 32.10 | 16.58 | 0.9229 |
| | CMS [30] | 4.28 | 36.52 | 14.88 | 0.9517 |
| | DIP [21] | 9.56 | 29.67 | 15.05 | 0.9117 |
| | MHF-net [25] | 2.62 | 40.26 | 12.90 | 0.9528 |
| | YONG [15] | 3.12 | 38.73 | 9.63 | 0.9646 |
| | UAL | 1.85 | 43.23 | 6.72 | 0.9862 |
| k_2 | NSSR [8] | 8.92 | 29.77 | 18.88 | 0.8445 |
| | CMS [30] | 4.92 | 34.75 | 17.15 | 0.8960 |
| | DIP [21] | 11.41 | 28.03 | 10.79 | 0.8960 |
| | MHF-net [25] | 2.84 | 39.53 | 13.43 | 0.9438 |
| | YONG [15] | 3.25 | 38.42 | 9.89 | 0.9621 |
| | UAL | 2.01 | 42.72 | 6.78 | 0.9859 |
| k_3 | NSSR [8] | 7.10 | 31.71 | 18.11 | 0.8661 |
| | CMS [30] | 4.29 | 35.90 | 16.67 | 0.9040 |
| | DIP [21] | 8.14 | 30.99 | 9.57 | 0.9245 |
| | MHF-net [25] | 2.81 | 39.62 | 13.47 | 0.9437 |
| | YONG [15] | 3.16 | 38.63 | 10.00 | 0.9613 |
| | UAL | 1.86 | 43.22 | 6.70 | 0.9862 |
| k_4 | NSSR [8] | 7.93 | 30.77 | 18.41 | 0.8569 |
| | CMS [30] | 4.66 | 35.22 | 16.88 | 0.8999 |
| | DIP [21] | 9.64 | 29.47 | 9.99 | 0.9126 |
| | MHF-net [25] | 2.84 | 39.53 | 13.43 | 0.9438 |
| | YONG [15] | 3.22 | 38.53 | 9.86 | 0.9608 |
| | UAL | 1.88 | 43.14 | 6.75 | 0.9860 |

6.06db. To show the effect of the pre-trained fusion module, we learn the whole SR network during unsupervised learning and show the result as 'UAL w/o pre-train' in Table 1. As can be seen, no pre-training results in obvious over-fitting, *e.g.* PSNR drops 5.72db. In addition, to demonstrate the necessity of fixing the pre-trained fusion module, we fine-tune it during unsupervised learning and show the result as 'UAL (fine-tune)' in Table 1, which also incurs obvious performance drop. Finally, we utilize the pre-defined $\hat{\mathbf{H}}$ instead of estimating \mathbf{H} during unsupervised learning to show the effect of degeneration estimation. Since incorrect demonstrate $\hat{\mathbf{H}}$ misleads learning adaptation module, the performance also drops, shown as 'UAL ($\hat{\mathbf{H}}$)' in Table 1.

Effect of components in fusion module The proposed fusion module contains two key components: the multi-branch structure and the mutual-guiding block. To illustrate the effect of the multi-branch structure, we conduct two experiments. The first removes the branch that takes the concatenation $[\mathbf{X}; \mathbf{Y}]$ as input, while the second keeps the concatenation branch but removes the other two. Their results are given as 'UAL (only concat)' and 'UAL w/o concat' in Table 1. Compared with the proposed UAL, their performance declines obviously. To demonstrate the effect of the mutual-guiding block, we remove them from the fusion module. The results are given as 'UAL w/o mutual-guiding' in Table 1. As can be seen, including the mutual-guiding blocks can improve the performance by a clear margin.

Effect of components in adaptation module In the adaptation module, both residual structure and self-guiding block are crucial. To verify this, we compare UAL with two variants, *i.e.* 'UAL w/o residual' and 'UAL w/o self-guiding' which remove the residual structure and the self-guiding block, respectively, from the adaptation module. Their results are given in Table 1. As can be seen, removing

Table 5. Performance when SR scale s is 32 and the degeneration is produced by k_1 and 40db noise.

| Methods | CAVE | | | | Havard | | | | NTIRE | | | |
|------------------------------|-------------|--------------|-------------|---------------|-------------|--------------|-------------|---------------|-------------|--------------|-------------|---------------|
| | RMSE | PSNR | SAM | SSIM | RMSE | PSNR | SAM | SSIM | RMSE | PSNR | SAM | SSIM |
| NSSR [8] | 9.05 | 30.00 | 16.47 | 0.9194 | 4.38 | 35.82 | 3.92 | 0.9620 | 3.34 | 38.45 | 2.28 | 0.9791 |
| CMS [30] | 4.57 | 35.50 | 11.06 | 0.9505 | 2.84 | 39.37 | 3.92 | 0.9680 | 2.03 | 42.57 | 1.79 | 0.9911 |
| DIP [21] | 9.90 | 28.83 | 17.34 | 0.9098 | 12.11 | 26.96 | 13.73 | 0.8502 | 13.18 | 26.11 | 12.30 | 0.8709 |
| MHF-net [25] | 3.72 | 37.65 | 13.46 | 0.9459 | 2.50 | 40.63 | 4.15 | 0.9763 | 1.98 | 43.26 | 1.90 | 0.9955 |
| YONG [15] | 4.98 | 34.83 | 12.84 | 0.9422 | 3.13 | 38.72 | 4.47 | 0.9678 | 7.57 | 32.83 | 4.22 | 0.9586 |
| UAL ($s=8$) | 2.89 | 39.67 | 9.24 | 0.9769 | 2.35 | 41.16 | 3.22 | 0.9788 | 1.71 | 43.76 | 1.68 | 0.9962 |
| UAL ($s=32$) | 2.78 | 40.06 | 7.78 | 0.9810 | 2.30 | 41.35 | 3.41 | 0.9785 | 1.57 | 44.38 | 1.58 | 0.9965 |
| UAL ($s=32 + \text{init}$) | 2.66 | 40.43 | 7.62 | 0.9831 | 2.14 | 41.82 | 3.30 | 0.9787 | 1.54 | 44.56 | 1.52 | 0.9967 |

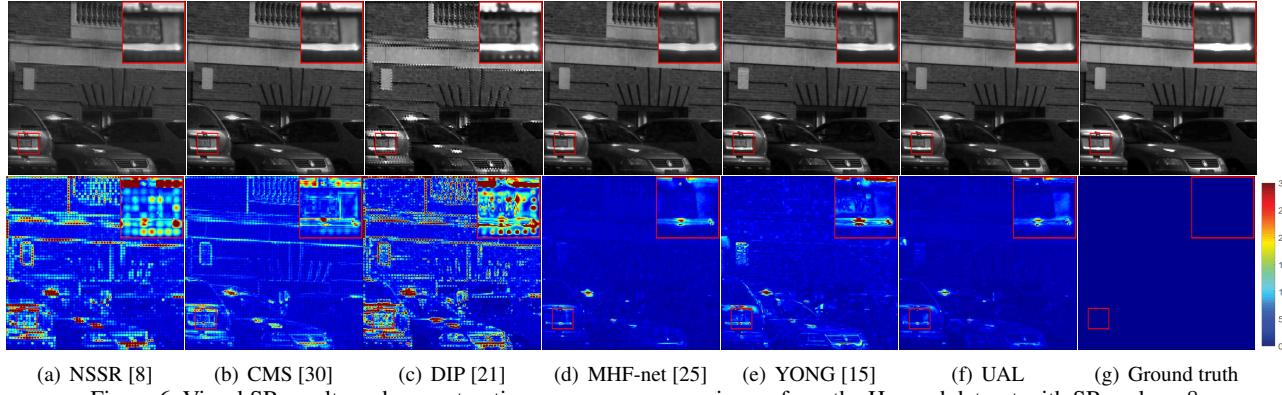


Figure 6. Visual SR results and reconstruction error maps on one image from the Harvard dataset with SR scale $s=8$.

either component causes a drop in performance.

4.3. Generalizability to unknown degenerations

We test the proposed UAL under various unknown degenerations on the CAVE dataset with SR scale $s=8$.

Degeneration with different kernels We utilize four kernels shown in Figure 5 and a fixed level (*e.g.*, 40db) of noise corruption to produce different degenerations \mathbf{H} . For each \mathbf{H} , we generate corresponding observed images, *e.g.* \mathbf{X} and \mathbf{Y} , and then feed them into all SR methods. The numerical results are reported in Table 2. As can be seen, since the degeneration $\hat{\mathbf{H}}$ predefined in NSSR [8], CMS [30], DIP [21] and MHF-net [25] deviates from the real \mathbf{H} , these methods fail to produce pleasing SR results. Although YONG [15] also estimates the degeneration, the shallow image prior limits its generalization performance. In contrast, the proposed UAL simultaneously conducts degeneration estimation and SR with deep image-specific image prior. Thus, it surpasses all competitors by a clear margin. This can be further supported by the visual results in Figure 1.

Degeneration with different noise levels We fix k_1 as kernel and introduce three different levels (*e.g.* 30db, 35db, 40db) of noise corruption to produce \mathbf{H} . The numerical results of all methods are reported in Table 3. Since most existing methods assume a clean LR HSI, they are sensitive to noise corruption. The proposed UAL can mitigate this problem by degeneration estimation, *e.g.* when the noise level is 35db, UAL improves the PSNR score by at least 3.42db.

4.4. Generalizability to various datasets

In this section, we first test the proposed UAL on different datasets and SR scales, when the degeneration \mathbf{H} is produced by kernel k_1 under 40db of noise corruption. Then, we test UAL on a real dataset.

Generalizability to different datasets To demonstrate the generalizability of the proposed UAL to various datasets, we obtain three versions of UAL, *i.e.*, 'UAL(CAVE)', 'UAL(Harvard)' and 'UAL(NTITRE)', in which the fusion module is trained on the training set of the CAVE, Harvard and NTIRE datasets, respectively. For fair comparison, we also obtain three versions of MHF-net [25] accordingly. Their numerical results are shown in Table 4. As can be seen, when generalizing to different datasets, the proposed UAL outperforms other competitors in most cases, *e.g.*, on CAVE dataset, 'UAL(Harvard)' improves the PSNR by at least 1.78db. Moreover, when training and testing on the same dataset, the performance of UAL is further improved. Thus, we can conclude that the proposed UAL generalizes better than other competitors to different datasets. This is further demonstrated by the visual SR results in Figure 6³.

Generalizability to different scales We further test all methods on the three datasets with SR scale $s=32$. To verify the generalizability of UAL to different scales, we train the fusion module with three different settings, termed 'UAL($s=8$)', 'UAL($s=32$)' and 'UAL($s=32+\text{init}$)'. 'UAL($s=8$)' and 'UAL($s=32$)' train the fusion module with

³In the followings, without special illustration, UAL denotes the version that trains the fusion module on CAVE dataset.

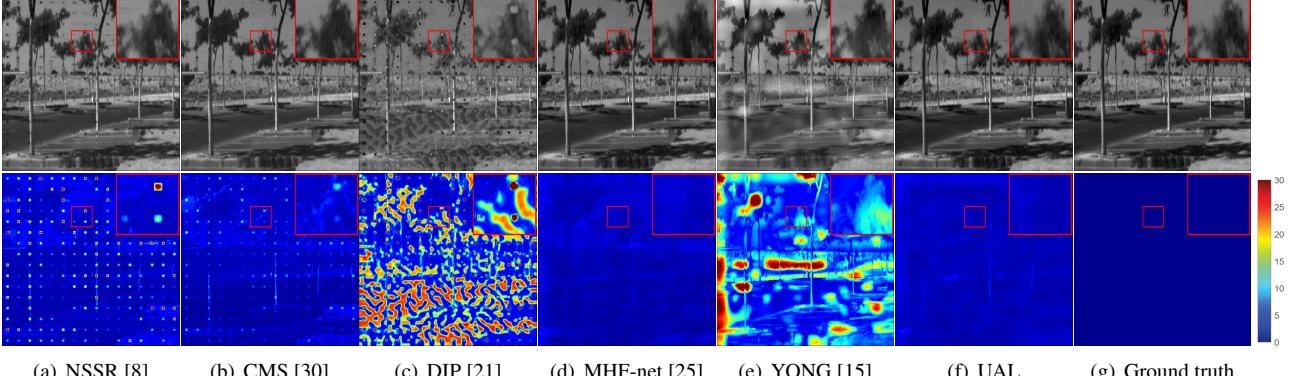


Figure 7. Visual SR results and reconstruction error maps on one image from the NTIRE dataset with SR scale $s=32$.

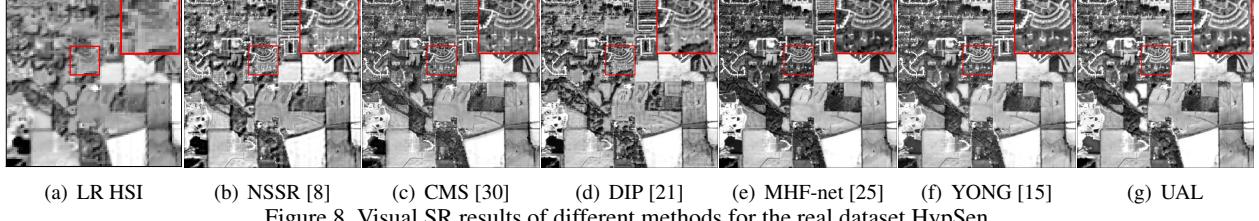


Figure 8. Visual SR results of different methods for the real dataset HypSen.

training pairs generated at scale $s=8$ and $s=32$, respectively. 'UAL($s=32+init$)' initializes the fusion module in 'UAL($s=32$)' with that from 'UAL($s=8$)'. The numerical results are reported in Table 5. As can be seen, despite not being trained on the test scale, 'UAL($s=8$)' still outperforms other competitors in most cases. When training on the test scale and being initialized, the performance of UAL can be further improved. A visual evidence is given in Figure 7.

Performance on real dataset Finally, we also test UAL on the real HypSen dataset [27]. With the spectral response function from [27], all methods are performed for SR by fusing the observed LR HSI and the HR MSI. Since there is no ground truth for the latent HR \mathbf{Z} , we follow the evaluation in [27]. In particular, we visualize the SR results of all methods in Figure 8. In addition, we also adopt a non-reference image quality evaluation metric [26] to evaluate the SR results of all methods, as shown in Table 6. We find that the proposed UAL recovers more details and obtain the best image quality.

4.5. Runtime analysis

We compute the average runtime of all methods on each image from the CAVE dataset when SR scale s is 8 and the degeneration is produced by the kernel \mathbf{k}_1 under 40db of noise corruption. All non-deep methods run on a workstation with 40 Intel Xeon CPU E5-2640 v4@2.40GHz, while deep learning based methods run on a Tesla V100 GPU. As shown in Table 7, the runtime of UAL is intermediate, and it runs much faster than most unsupervised competitors. Moreover, in practice, we can make a trade-off between performance and efficiency for the proposed UAL by tuning the optimization epochs of unsupervised learning.

Table 6. No-reference HSI quality measurement score [26] of each method on the real HSI SR task. (The smaller the better)

| Methods | NSSR [8] | CMS [30] | DIP [21] | MHF-net [25] | YONG [15] | UAL |
|---------|----------|----------|----------|--------------|-----------|--------------|
| Score | 12.84 | 11.59 | 16.36 | 11.19 | 12.47 | 10.98 |

Table 7. Average runtime on CAVE dataset when SR scale s is 32 and the degeneration is produced by \mathbf{k}_1 with 40db noise.

| Methods | NSSR [8] | CMS [30] | DIP [21] | MHF-net [25] | YONG [15] | UAL |
|----------|----------|----------|----------|--------------|-----------|-------|
| Time (s) | 104.69 | 301.71 | 428.21 | 1.31 | 8.36 | 44.21 |

5. Conclusion

In this study, we present an UAL framework for fusion-based HSI SR, which shows good generalization performance on different datasets and unknown degenerations. The model benefits from implicitly learning a deep image-specific prior as well as estimating the unknown degeneration. To this end, we first develop a two-stage SR network and pretrain the first stage in a supervised manner. Then, we introduce a simple degeneration network to assist learning the second stage for SR and the degeneration directly from the specific pair of observed images in unsupervised learning. Extensive experiments demonstrate the effectiveness of the proposed method. Moreover, it provides a promising way to generalize deep models to unseen cases in practice, which can be applied to other image restoration tasks, such as image denoising, deblurring etc.

6. Acknowledgement

This work was supported in part by the National Natural Science Foundation of China(No. 61671385), in part by the Science, Technology and Innovation Commission of Shenzhen Municipality(No.JCYJ20190806160210899).

References

- [1] Naveed Akhtar and Ajmal S Mian. Hyperspectral recovery from rgb images using gaussian processes. *IEEE transactions on pattern analysis and machine intelligence*, 2018.
- [2] Naveed Akhtar, Faisal Shafait, and Ajmal Mian. Sparse spatio-spectral representation for hyperspectral image super-resolution. In *European Conference on Computer Vision*, pages 63–78. Springer, 2014.
- [3] Naveed Akhtar, Faisal Shafait, and Ajmal Mian. Bayesian sparse representation for hyperspectral image super resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3631–3640, 2015.
- [4] Naveed Akhtar, Faisal Shafait, and Ajmal Mian. Hierarchical beta process with gaussian process prior for hyperspectral image super resolution. In *European Conference on Computer Vision*, pages 103–120. Springer, 2016.
- [5] Rushil Anirudh, Jayaraman J Thiagarajan, Bhavya Kailkhura, and Timo Bremer. An unsupervised approach to solving inverse problems using generative adversarial networks. *arXiv preprint arXiv:1805.07281*, 2018.
- [6] Ayan Chakrabarti and Todd Zickler. Statistics of real-world hyperspectral images. In *CVPR 2011*, pages 193–200. IEEE, 2011.
- [7] Renwei Dian, Leyuan Fang, and Shutao Li. Hyperspectral image super-resolution via non-local sparse tensor factorization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5344–5353, 2017.
- [8] Weisheng Dong, Fazuo Fu, Guangming Shi, Xun Cao, Jin-jian Wu, Guangyu Li, and Xin Li. Hyperspectral image super-resolution via non-negative structured sparse representation. *IEEE Transactions on Image Processing*, 25(5):2337–2352, 2016.
- [9] Netalee Efrat, Daniel Glasner, Alexander Apartsin, Boaz Nadler, and Anat Levin. Accurate blur models vs. image priors in single image super-resolution. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2832–2839, 2013.
- [10] Charilaos I Kanatsoulis, Xiao Fu, Nicholas D Sidiropoulos, and Wing-Kin Ma. Hyperspectral super-resolution via coupled tensor factorization: Identifiability and algorithms. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3191–3195. IEEE, 2018.
- [11] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- [12] Charis Lanaras, Emmanuel Baltsavias, and Konrad Schindler. Hyperspectral super-resolution by coupled spectral unmixing. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3586–3594, 2015.
- [13] Jiaojiao Li, Xi Zhao, Yunsong Li, Qian Du, Bobo Xi, and Jing Hu. Classification of hyperspectral imagery using a new fully convolutional neural network. *IEEE Geoscience and Remote Sensing Letters*, 15(2):292–296, 2018.
- [14] Shutao Li, Renwei Dian, Leyuan Fang, and José M Bioucas-Dias. Fusing hyperspectral and multispectral images via coupled sparse tensor factorization. *IEEE Transactions on Image Processing*, 27(8):4118–4130, 2018.
- [15] Yong Li, Lei Zhang, Chunna Tian, Chen Ding, Yanning Zhang, and Wei Wei. Hyperspectral image super-resolution extending: An effective fusion based method without knowing the spatial transformation matrix. In *2017 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1117–1122. IEEE, 2017.
- [16] Vinod Nair and Geoffrey E Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pages 807–814, 2010.
- [17] Nasser M Nasrabadi. Hyperspectral target detection: An overview of current and future challenges. *IEEE Signal Processing Magazine*, 31(1):34–44, 2013.
- [18] Yi Peng, Deyu Meng, Zongben Xu, Chenqiang Gao, Yi Yang, and Biao Zhang. Decomposable nonlocal tensor dictionary learning for multispectral image denoising. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2949–2956, 2014.
- [19] Ying Qu, Hairong Qi, and Chiman Kwan. Unsupervised sparse dirichlet-net for hyperspectral image super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2511–2520, 2018.
- [20] Radu Timofte, Shuhang Gu, Jiqing Wu, and Luc Van Gool. Ntire 2018 challenge on single image super-resolution: Methods and results. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 852–863, 2018.
- [21] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Deep image prior. In *2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.
- [22] Hien Van Nguyen, Amit Banerjee, and Rama Chellappa. Tracking via object reflectance using a hyperspectral video camera. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops*, pages 44–51. IEEE, 2010.
- [23] Xintao Wang, Ke Yu, Chao Dong, and Chen Change Loy. Recovering realistic texture in image super-resolution by deep spatial feature transform. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 606–615, 2018.
- [24] Eliot Wycoff, Tsung-Han Chan, Kui Jia, Wing-Kin Ma, and Yi Ma. A non-negative sparse promoting algorithm for high resolution hyperspectral imaging. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pages 1409–1413. IEEE, 2013.
- [25] Qi Xie, Minghao Zhou, Qian Zhao, Deyu Meng, Wangmeng Zuo, and Zongben Xu. Multispectral and hyperspectral image fusion by ms/hs fusion net. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1585–1594, 2019.
- [26] Jingxiang Yang, Yongqiang Zhao, Chen Yi, and Jonathan Cheung-Wai Chan. No-reference hyperspectral image quality assessment via quality-sensitive features learning. *Remote Sensing*, 9(4):305, 2017.

- [27] Jingxiang Yang, Yong-Qiang Zhao, and Jonathan Chan. Hyperspectral and multispectral image fusion via deep two-branches convolutional neural network. *Remote Sensing*, 10(5):800, 2018.
- [28] Fumihito Yasuma, Tomoo Mitsunaga, Daisuke Iso, and Shree K Nayar. Generalized assorted pixel camera: post-capture control of resolution, dynamic range, and spectrum. *IEEE transactions on image processing*, 19(9):2241–2253, 2010.
- [29] Haichao Zhang, Yanning Zhang, Haisen Li, and Thomas S Huang. Generative bayesian image super resolution with natural image prior. *IEEE Transactions on Image processing*, 21(9):4054–4067, 2012.
- [30] Lei Zhang, Wei Wei, Chengcheng Bai, Yifan Gao, and Yanning Zhang. Exploiting clustering manifold structure for hyperspectral imagery super-resolution. *IEEE Transactions on Image Processing*, 27(12):5969–5982, 2018.
- [31] Lei Zhang, Wei Wei, Yanming Zhang, Chunhua Shen, Anton van den Hengel, and Qinfeng Shi. Cluster sparsity field: An internal hyperspectral imagery prior for reconstruction. *International Journal of Computer Vision*, 126(8):797–821, 2018.
- [32] Lei Zhang, Wei Wei, Yanning Zhang, Chunna Tian, and Fei Li. Reweighted laplace prior based hyperspectral compressive sensing for unknown sparsity. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2274–2281, 2015.
- [33] Maria Zontak and Michal Irani. Internal statistics of a single natural image. In *CVPR 2011*, pages 977–984. IEEE, 2011.