

RGB-Guided Hyperspectral Image Upsampling

Hyeokhyen Kwon
 KAIST

hyeokhyen@kaist.ac.kr

Yu-Wing Tai
 SenseTime Group Limited
 yuwing@gmail.com

Abstract

Hyperspectral imaging usually lack of spatial resolution due to limitations of hardware design of imaging sensors. On the contrary, latest imaging sensors capture a RGB image with resolution of multiple times larger than a hyperspectral image. In this paper, we present an algorithm to enhance and upsample the resolution of hyperspectral images. Our algorithm consists of two stages: spatial upsampling stage and spectrum substitution stage. The spatial upsampling stage is guided by a high resolution RGB image of the same scene, and the spectrum substitution stage utilizes sparse coding to locally refine the upsampled hyperspectral image through dictionary substitution. Experiments show that our algorithm is highly effective and has outperformed state-of-the-art matrix factorization based approaches.

1. Introduction

Hyperspectral imaging acquires spectral characteristics of a scene through capturing a number of contiguous narrow band images. It has been a promising tool for applications in geosensing, cultural heritage, and various computer vision tasks [30, 21, 13]. Due to the hardware design limitations [9], hyperspectral sensors often have a low spatial resolution, but very fine spectral resolution. In addition, due to reduced incoming illumination by narrow band filters, it usually has low signal-to-noise ratio. Consequently, a hyperspectral image is noisy if a long exposure time is not guaranteed.

Compared with hyperspectral sensors, latest imaging sensors capture a RGB image with resolution of multiple times larger than a hyperspectral image and with higher signal-to-noise ratio given the same exposure time. Because of these attractive properties, various approaches have been proposed to fuse a low-resolution hyperspectral image and a high-resolution RGB (or gray scale) image to reconstruct a high-resolution hyperspectral image as demonstrated in [12, 2, 9]. In this paper, we follow this direction and introduce a new algorithm which outperforms state-of-the-art approaches for hyperspectral image upsampling.

Our algorithm consists of two stages: spatial upsampling stage and spectrum substitution stage. The spatial upsampling stage increases the spatial resolution of a hyperspectral image through the guidance of the high resolution RGB image. This is achieved by estimating an optimal linear combination of exemplar patches for super-resolution reconstruction. The exemplar patches couple hyperspectral and RGB images with identical structures learnt from training examples. The spatial upsampling stage not only enhance spatial resolution, but also reduce image noise in a hyperspectral image. In the spectrum substitution stage, we assume a local region of a scene contains limited number of materials, and adopt the sparse coding [16] to learn a local spectrum dictionary. We evaluate the upsampled hyperspectral image after the first stage using the learnt local spectrum dictionary. The optimal linear combination of spectrum basis functions with sparse coefficients which best describes the upsampled hyperspectral and RGB images is used to reconstruct our final output. In short, the first stage estimates a proxy solution which *unmix* the hyperspectrum observation within a pixel using the guidance from a high resolution RGB image, and the second stage refines the spectrum of the proxy solution based on the limited materials assumption within a local region of a scene.

We have evaluated our proposed algorithm extensively using real world hyperspectral image datasets. The experiments show that our algorithm is highly effective, and has outperformed state-of-the-art matrix factorization based approaches both qualitatively and quantitatively.

2. Related work

Hyperspectral imaging was first introduced as an application in astronomy and satellite imaging. It aims to capture spectrum power distribution of electromagnetic wavelengths ranged from infrared spectrum to ultraviolet spectrum. Since certain materials have unique spectral signature, it has been very useful in material, agriculture, chemical, and environment analysis [9].

Since hyperspectral images have limited spatial resolution, various techniques have been proposed to increase resolution of a hyperspectral image. A common tech-

nique in remote sensing is the pan-sharpening [3] which merges a high resolution panchromatic (gray scale) image and a low resolution hyper-/multi-spectral image to reconstruct a high resolution hyper-/multi-spectral image. Representative techniques in pan-sharpening including the Intensity-Hue-Saturation (IHS) transform [4], PCA-based method [22], Gramm-Schmidt algorithm [15], and wavelet-based method [20]. Although some good results have been demonstrated, as discussed in [12], the single channel panchromatic image is too limited in spectral resolution. Consequently, the reconstructed high resolution hyperspectral images contain spectral distortions.

In image processing, filtering techniques such as joint bilateral upsampling [14] and guided image filter [10] have been proposed to enhance image resolution. Their key idea is to borrow high resolution edges from another image of the same scene to guide the filtering process. These techniques process each channel individually and often over smooth image structures.

Recent approaches by Kawakami *et al.* [12] and Huang *et al.* [11] utilize the matrix factorization to learn a global basis functions of spectral signals. By utilizing the corresponding high resolution RGB image, sparse coefficient of spectral basis functions which best reconstruct the RGB signals can be uniquely determined. Finally, the high resolution hyperspectral image is reconstructed by using the estimated coefficient and the spectral basis functions to reconstruct the hyperspectral signals. Since the matrix factorization based techniques achieved the state-of-the-art performance, several follow up works have been proposed [24, 29, 28]. Wycoff *et al.* [24] proposed alternating optimization for non-negative sparse matrix factorization. Yokoya *et al.* [28] proposed coupled non-negative matrix factorization. Akhtar *et al.* [2] additionally consider spatial structure of high resolution RGB images. A major limitation of the matrix factorization based approaches is that the estimated sparse coefficient is only determined by the high resolution RGB observations. When different materials with the same or very similar RGB observations presented in the same scene, these techniques often fail to distinguish the different materials and thus the reconstructed hyperspectral images are inaccurate.

3. Proposed Algorithm

Let $\mathbf{H} \in \mathbb{R}^{W \times H \times S}$ denote a high resolution hyperspectral image, where W , H and S are the image width, the image height, and the number of sampled wavelengths respectively. Our goal is to estimate \mathbf{H} , given $\mathbf{L} \in \mathbb{R}^{w \times h \times S}$, a low resolution hyperspectral image where $w \ll W$ and $h \ll H$, and $\mathbf{C} \in \mathbb{R}^{W \times H \times 3}$, a high resolution RGB image as inputs. We assume \mathbf{L} and \mathbf{C} are well aligned. We first present our spatial upsampling method, followed by the spectrum substitution method.

3.1. Spatial Upsampling

To upsample \mathbf{L} , we adopt a fast learning-based single image super-resolution algorithm [25] with modifications to account for the high resolution structures in \mathbf{C} , and spectrum correlation across different wavelength channels in \mathbf{L} . The approach by [25] represents the state-of-the-art single image super-resolution algorithm with fast running time.

Learning the HR-LR exemplar In order to learn the HR-LR (High Resolution-Low Resolution) exemplars for learning based single image super-resolution, we prepare training examples by downsampling a high resolution hyperspectral image using bicubic interpolation to create a synthetic low resolution hyperspectral image. The corresponding RGB image is given in the training examples. For example, the CAVE hyperspectral image dataset [27] provides high resolution hyperspectral images and the corresponding RGB images in pairs. In here, we assume the sampled wavelengths in training examples are identical to the sampled wavelengths in the target hyperspectral image.

We sample image patches from training examples, each patch is size of 5×5 , and with corresponding primitive structures in \mathbf{H} , \mathbf{L} and \mathbf{C} denote as \mathcal{P}_H , \mathcal{P}_L and \mathcal{P}_C respectively. The resolution ratio between \mathbf{H} and \mathbf{L} in the training examples is 2. If the upsampling factor (resolution ratio between \mathbf{C} and \mathbf{L}) is larger than 2, we upsample the target \mathbf{L} multiple times. After each $2 \times$ spatial upsampling, we apply the spectrum substitution, and repeat the processes until \mathbf{L} meets the target resolution.¹ For each sampled patch, \mathcal{P} , we compute its Y (luminance) channel image using its RGB values, and then apply mean subtraction to create a feature vector by stacking pixels of the processed Y channel image. The sampled patches are then clustered into K groups according to the feature vectors. In our implementation, we used the kmeans++ offered in the VLFeat open source library [23] for clustering. The number of clusters is set to $K = 4096$, and there are around 400 patches within each cluster. The clustered patches in each group are then served as exemplars for the super-resolution purpose. Figure 1 shows some of clustered exemplars.

HR hyperspectral image reconstruction Let $\mathcal{S} = \{\mathcal{S}_H(k, q), \mathcal{S}_L(k, q), \mathcal{S}_C(k, q)\}, 1 \leq k \leq K, 1 \leq q \leq Q(k)$ denote the trained exemplars, where K is the number of clustered group, and $Q(k)$ is number of exemplars in each group. Because exemplars in each group share very similar primitive structures, we assume that each exemplar in each group can be represented as a linear combination of other exemplars within the same group, *i.e.*, $\mathcal{S}(k, q) \doteq \sum_{p=1, p \neq q}^{Q(k)} \varphi_p \mathcal{S}(k, p)$, where $\{\varphi_p\}$ is the linear

¹The target resolution depends on the hardware setting. In our synthetic experiments, we follow previous works to set the target resolution as $32 \times$. However, in the real world experiments, we can change it to other scaling factor.

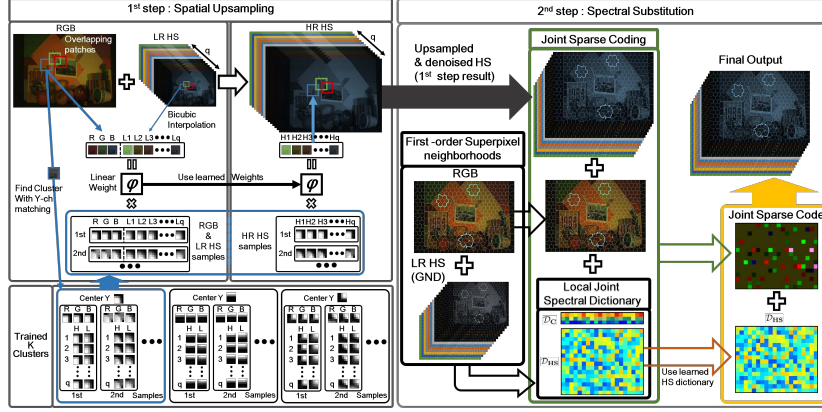


Figure 1: Overview of our algorithm. In the first spatial upsampling step, overlapping patches from input images are first matched to appropriate membership by minimizing the L_2 -distance between Y -channel to find the closest structure. Next, linear weights are learned from the RGB-LR hyperspectral(HS) input and the corresponding subspace samples. The learned weights are used to reconstruct target HR HS patch with HR HS subspace samples. In the second spectral substitution step, for each first-order superpixel neighbourhood, we learn a local RGB-LR HS joint spectral dictionary. With the learned dictionary, RGB-HR HS joint sparse coding is obtained. Finally, the target local HR HS spectrum is reconstructed with the locally learned HS dictionary and the joint sparse code.

coefficients. Based on this assumption, when a new testing patch comes, cluster membership of an input patch is determined by minimizing the L_2 -distance between Y channel of cluster center and input patch. And then, we can estimate the linear coefficients based on the observed \mathcal{P}_L and \mathcal{P}_C which minimizes:

$$\varphi^* = \arg \min_{\varphi} \left\| \begin{bmatrix} \mathcal{P}_L \\ \mathcal{P}_C \end{bmatrix} - \sum_{p=1}^{Q(k)} \varphi_p \begin{bmatrix} \mathcal{S}_L(k, p) \\ \mathcal{S}_C(k, p) \end{bmatrix} \right\|_2^2. \quad (1)$$

The optimal solution, φ^* , can be obtained by a simple linear regression. Using φ^* , \mathcal{P}_H is reconstructed as:

$$\mathcal{P}_H = \sum_{p=1}^{Q(k)} \varphi_p^* \mathcal{S}_H(k, p). \quad (2)$$

After we estimate \mathcal{P}_H , we add the patch mean of \mathbf{L} to get the correct upsampled \mathbf{H} . In our implementation, the reconstructed patches were accurate enough, such that we simply averaged the overlapping patch area to enhance local compatibility, inspired by Chang *et al.* [6].

Since the input \mathbf{L} can be noisy, we additionally incorporate a structure guided total variation regularization [18, 8]:

$$\mathbf{H}^\circ = \arg \min_{\mathbf{H}} \|\mathbf{H} - \hat{\mathbf{H}}\|_2^2 + \mu(1 - |\nabla_m \mathbf{C}|) |\nabla \mathbf{H}|_1, \quad (3)$$

where $\hat{\mathbf{H}}$ is the solution after the exemplar super-resolution, $\nabla_m \mathbf{C} = \max(|\nabla_R \mathbf{C}|, |\nabla_G \mathbf{C}|, |\nabla_B \mathbf{C}|)$ is the maximum absolute gradient of \mathbf{C} across the RGB channels, $|\nabla \mathbf{H}|_1$ is the total variation regularization, and $\mu = 0.01$ is the regularization weight. This regularization weight is fixed in all of our experiments. By including the structure weight from $(1 - |\nabla_m \mathbf{C}|)$, we can effectively avoid over-smoothing in large gradient regions in \mathbf{C} . Each channel in \mathbf{H} is processed individually for Equation (3).

Discussion Here, we compare our spatial upsampling method with single image super-resolution, especially [25]. Our method utilizes \mathcal{S}_C as a guidance to search the optimal linear coefficients which allows us to achieve higher accuracy reconstruction than using \mathcal{S}_L alone. In addition, by searching the same linear coefficients across different channels, we can preserve spectrum fidelity while the approach in [25] process each color channel individually. Consequently, results by [25] contains distortion in spectrum.

Our spatial upsampling method already can achieve high quality upsampling results. However, we notice that there are still minor errors in the upsampled result from Equation (2), the artificial pixels and distorted spectrum. One possible reason is that the hyperspectrum of materials and specific structural resolution in the target image may not exist in the training examples. Although we obtain the approximate spectrum by finding the optimal linear combination of material spectrum in training examples that best match with the low resolution observations, certain amount of spectrum distortions presented after the upsampling. For defective pixels, we could repair them through Equation (3) effectively. For distorted spectrum, we present the spectrum substitution method which further enhances our results in the next sub-section.

3.2. Spectrum Substitution

Our spectrum substitution utilizes the sparse coding to learn a local spectrum dictionary, and sparse linear coefficients which will be used to substitute the upsampled results from the previous stage for spectrum refinement. This process follows the assumption by [12] in using the matrix factorization to learn spectrum basis functions for hyperspectral image reconstruction. The work by [12] assumes a scene contains limited materials, and therefore the hy-

perspectrum can be modelled by a linear combination of spectrum basis functions with sparse coefficients. In here, instead of making a global assumption about number of materials in a scene, we assume the number of materials within a local region is limited.

We segment the upsampled hyperspectral image into superpixels using the approach by [17]. For each superpixel, we collect its first order neighborhood (directly connected superpixels), and apply the sparse coding [16, 26] to learn a local dictionary:

$$\arg \min_{\mathcal{D}, \alpha} \left\| \begin{bmatrix} \mathbf{L}(\vec{x}) \\ \mathbf{C}^d(\vec{x}) \end{bmatrix} - \begin{bmatrix} \mathcal{D}_{\mathbf{HS}} \\ \mathcal{D}_{\mathbf{C}} \end{bmatrix} \alpha \right\|_2^2 + \lambda |\alpha|_1 \quad (4)$$

where $\vec{x} \in \mathbb{R}^{1 \times N}$ is index of pixels within a superpixel and its first order neighborhood, $\mathcal{D} = \{\mathcal{D}_{\mathbf{HS}}, \mathcal{D}_{\mathbf{C}}\} \in \mathbb{R}^{(S+3) \times N}$ is the learnt local dictionary which corresponds to the basis functions of local reflectance around a superpixel, $\alpha \in \mathbb{R}^{N \times 1}$ is the sparse coefficient, and N is number of involved pixels. We perform the training process in low resolution by downsampling \mathbf{C} to \mathbf{C}^d to match the resolution of \mathbf{L} . After we estimate \mathcal{D} , we drop the columns of \mathcal{D} if the corresponding coefficient in α is zero². Note that we utilize \mathbf{L} instead of \mathbf{H}° to learn the local dictionary since the spectrum in \mathbf{H}° may have been distorted.

Once we estimate \mathcal{D} , we can reconstruct \mathbf{H}^* using \mathbf{H}° , \mathbf{C} and sparsity as guides [16]:

$$\arg \min_{\omega} \left\| \begin{bmatrix} \mathbf{H}^\circ(\vec{x}) \\ \mathbf{C}(\vec{x}) \end{bmatrix} - \begin{bmatrix} \mathcal{D}_{\mathbf{HS}} \\ \mathcal{D}_{\mathbf{C}} \end{bmatrix} \omega \right\|_2^2 + \lambda |\omega|_1, \quad (5)$$

$$\mathbf{H}^*(\vec{x}) = \mathcal{D}_{\mathbf{HS}} \omega. \quad (6)$$

The spectrum substitution process is applied to each superpixel individually, with different \mathcal{D} for each superpixel. Since \mathcal{D} is learnt from overlapping superpixel after including first order neighborhood, there is no discontinuity artifact introduced after the spectral substitution. The parameter λ is set to 0.1 empirically and it is fixed in all of our experiments.

Discussion There are two major differences between our spectrum substitution method and the matrix factorization approach by [12]. First, our limited material assumption is a local assumption while in [12] is a global assumption. Thus, our assumption is more general. Second, when estimating the sparse coefficient, ω , in Equation (5), [12] uses only the RGB image, \mathbf{C} , as guides, while we also use our upsampled image, \mathbf{H}° , as guides. This not only allows us to estimate more accurate sparse coefficients, but also allows us to distinguish spectrum of materials in a scene whose RGB observations are identical. Since hyperspectrum to

RGB is a many-to-one process, using the RGB observations alone cannot distinguish such different materials. However, without an accurate proxy solution, the estimation of ω can be biased to a wrong solution. As will be demonstrated in our experiments, simply replace \mathbf{H}° by bicubic upsampling cannot achieve the same quality results as ours. The first stage and the second stage of our algorithm complement the strength of each other to achieve high quality results which outperforms state-of-the-art algorithms.

Compared with coupled dictionary approach for image super-resolution [26], we applied the coupled dictionary training in spectrum domain, instead of spatial domain. The method by [26] does not consider spectrum correlation during super-resolution. Since our method has guiding structures from high resolution RGB image, our approach performs better than just applying single image super-resolution to each channel individually.

4. Experiments

We tested our algorithm on the CAVE dataset [27]³, the Harvard dataset [5]⁴, the NUS dataset [19]⁵ and the Kawakami dataset [12]⁶.

The CAVE and the NUS datasets contain hyperspectral images ranged from 400nm to 700nm at every 10nm wavelength, the Harvard dataset contains hyperspectral images ranged from 420nm to 720nm at every 10nm, and the Kawakami dataset contains hyperspectral images ranged 415nm to 730nm at every 5nm. Since each dataset has its own wavelength features, we trained the exemplars for spatial upsampling separately for each dataset. The training and testing examples in each dataset are separated.

For all testing cases except for Kawakami dataset, the upsampling factor is $M = 32\times$. Thus, the number of level in our multi-level implementation is $5 = \lceil \log_2 M \rceil$. In the spatial training, the number of training samples was 10^6 for all cases. For spectral substitution, number of superpixels in segmentation [17] was $\{8, 20, 50, 150, 300\}$ at 1st to 5th level. To create the training set, we downsample captured HSI images using bicubic interpolation. For $32\times$ up-sampling, the same $2\times$ dictionary is used for all 5 levels.

The measurement metric in our experiments is the standard RMSE⁷ in 8 bit resolution. our reported RMSE values

³Cave dataset can be downloaded from <http://www.cs.columbia.edu/CAVE/databases/multispectral/>

⁴Harvard dataset can be downloaded from <http://vision.seas.harvard.edu/hyperspec/>

⁵http://www.comp.nus.edu.sg/~whitebal/spectral_reconstruction/

⁶We contacted the first author for their dataset.

⁷The standard RMSE used for evaluation in this paper, e.g. $RMSE(\theta) = \sqrt{E((\hat{\theta} - \theta)^2)}$ where $\hat{\theta}$ is the upsampled result and θ is ground-truth, is different from the metrics used in [2]. Comparisons with their error metrics are presented in the supplemental material.

²This method enhances generality of dictionary and avoids trivial solution by removing outliers

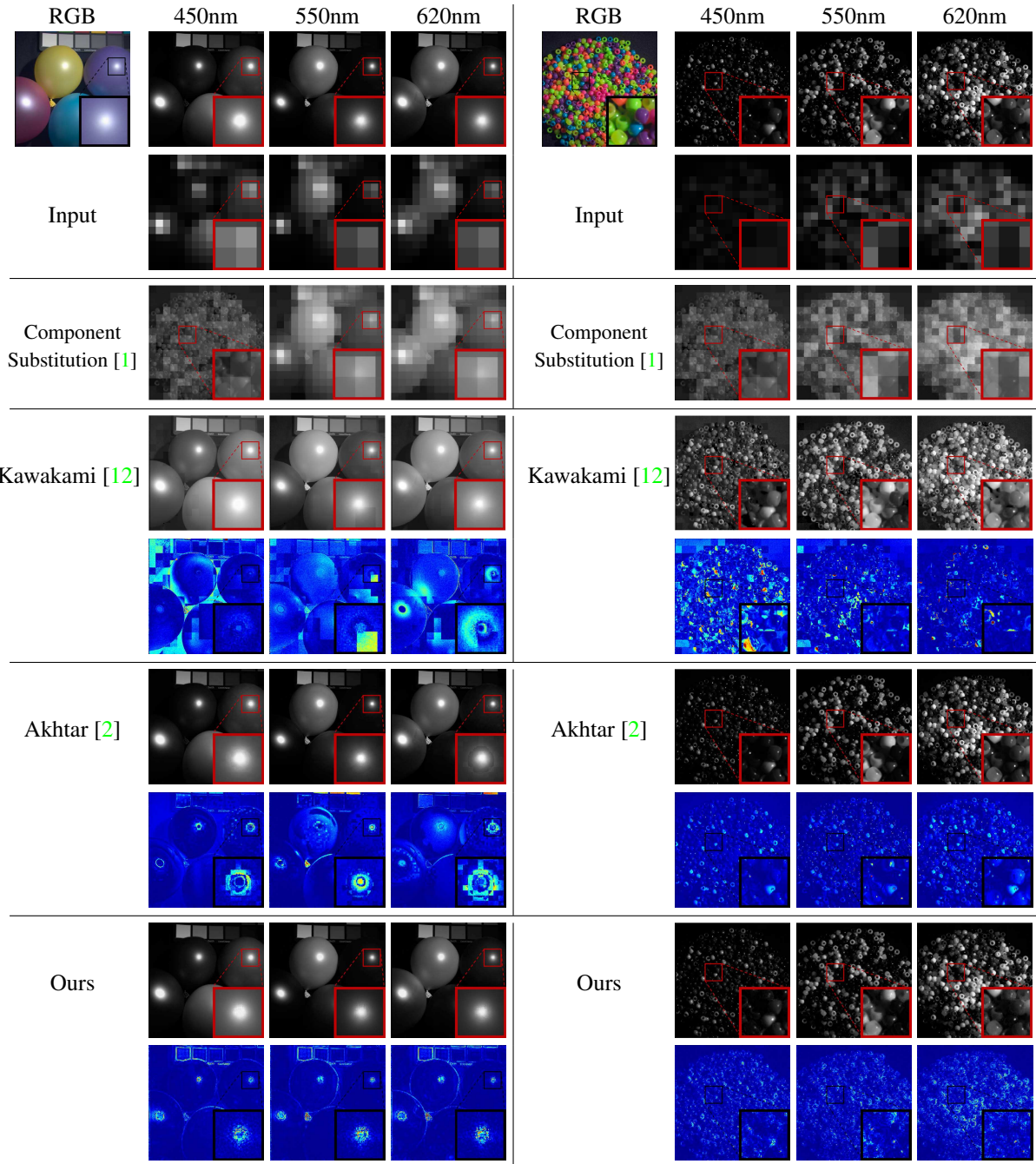


Figure 2: Comparisons on the CAVE dataset for 450nm, 550nm, and 620nm: Balloons(left) and Beads(right). The RGB and Ground-truth are shown in top row. Results and the corresponding error maps are shown in sequence.

of Akhtar *et al.* [2]⁸ was the test results from their source codes with their proposing parameters. For the other compared methods, Aiazzi *et al.* [1], Kawakami *et al.* [12], and Huang *et al.* [11], we referred reported values in [12] and [2], because we could not get their source codes. Due to space limitation, we presented only representative examples

⁸Akhtar *et al.*'s source code can be downloaded from <http://www.csse.uwa.edu.au/~ajmal/code/HSISuperRes.zip>

in the paper. More results are in the supplemental material.

4.1. Cave dataset

We have tested our algorithm on the CAVE dataset. The tested examples vary from planar to complex objects with various color and spectral configurations from diverse materials. Some of our results and comparisons are shown in Figure 2. In this dataset, the corresponding RGB image is

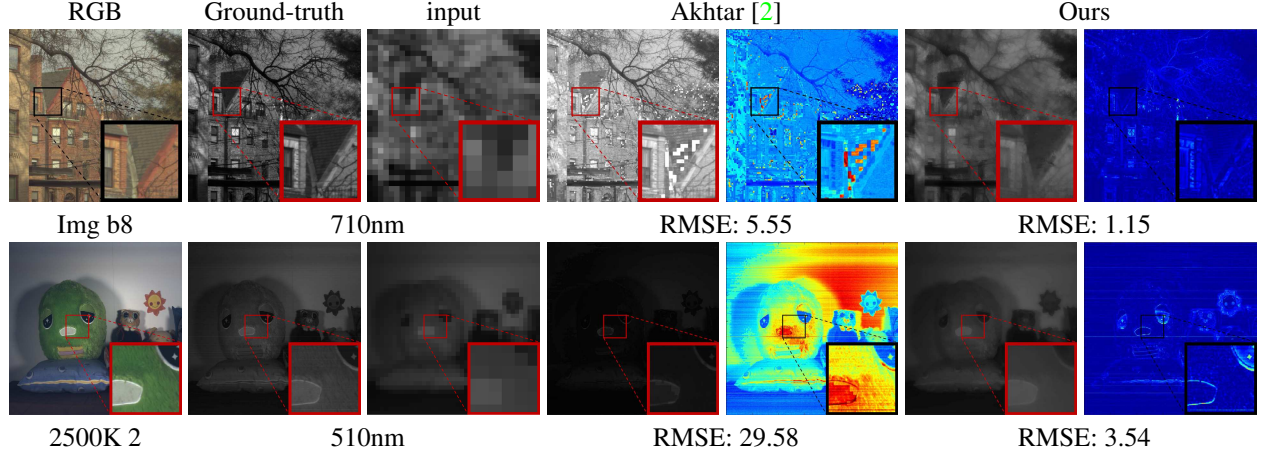


Figure 3: Comparisons on Harvard (Top) and NUS (Bottom) dataset.

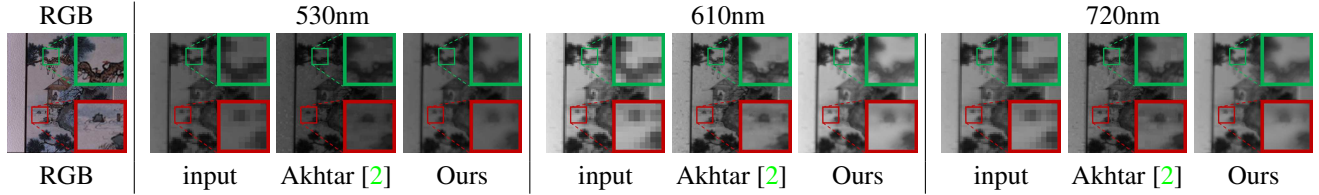


Figure 4: Comparison examples from Kawakami dataset.

Table 1: Upsampling of Cave dataset(32 \times).

	Balloons	Beads	Sponges	Paintings
Aiazzi <i>et al.</i> [1]	13.9	28.5	19.9	12.2
Kawakami <i>et al.</i> [12]	3.0	9.2	3.7	4.7
Akhtar <i>et al.</i> [2]	3.8	9.12	4.74	4.17
Ours	1.64	6.92	2.89	3.55
	Flowers	CD	Peppers	Face
Aiazzi <i>et al.</i> [1]	14.4	13.3	13.7	13.1
Kawakami <i>et al.</i> [12]	5.4	8.2	4.7	3.3
Akhtar <i>et al.</i> [2]	8.94	16.1	3.91	4.59
Ours	2.61	3.48	1.96	2.45

also given, but the spectral sensitivity function to convert hyperspectrum to RGB is not given. The dataset is relatively clean with high signal-to-noise ratio in comparisons with other datasets. The ground-truth resolution of hyperspectral image was 512×512 and our input resolution of hyperspectral image was 16×16 . Since camera spectra sensitivity is not given, we used the camera sensitivity function from Nikon D700 for producing results of Akhtar *et al.*, as presented in their experiment [2]. But for our case, we just used the given RGB image without the need of the spectral sensitivity function. Our work consistently outperformed the compared state-of-the-arts methods as shown in Table 1.

4.2. Harvard dataset

One tested example of the Harvard dataset is shown in Figure 3. The original image resolution of hyperspectral images given in the dataset is 1392×1040 , but we cropped

the images to resolution of 1024×1024 for our convenience of processing. This dataset does not provide corresponding RGB images. To provide a fair test setting, We used same RGB spectral response for both of Akhtar *et al.* [2] and ours. For the RGB spectral response, we have normalized the hyperspectral image with respect to the given camera sensitivity, and RGB channel image was generated by firstly converting the normalized hyperspectral image to CIE 1931 2-deg, XYZ⁹, converting CIE tristimulus value to sRGB, and applying gamma correction to sRGB values. After these processes¹⁰, we obtain the corresponding RGB image as inputs of our algorithm. The input resolution of hyperspectral image was 32×32 . We show the results of using both Nikon D700 response function and our estimated response function. Our work consistently showed the best results as shown in Table 2.

4.3. NUS dataset

Our approach was also tested on the NUS dataset and compared with Akhtar *et al.* [2]. One example is shown in Figure 3. In the tested data, the first example was taken under natural daylight and shade condition and the second to fifth were taken under illumination by a metal halide lamp. Temperature of the metal halide lamp for the second and

⁹Color Matching Function for CIE 1931 2-deg, XYZ can be downloaded from <http://www.cvrl.org/icons/datawe.gif>

¹⁰Details can be found at http://personalpages.manchester.ac.uk/staff/david.foster/Tutorial_HSI2RGB/Tutorial_HSI2RGB.html

Table 2: Upsampling of Harvard dataset($32\times$).

Method	Img 1	Img b5	Img b8	Img d4
Huang <i>et al.</i> [11]	4.3	2.6	7.6	4.0
Kawakami <i>et al.</i> [12]	3.9	2.8	6.9	3.6
Nikon D700 response				
Akhtar <i>et al.</i> [2]	2.0	1.4	4.1	1.5
Ours	0.72	0.75	1.56	0.53
Our estimated response				
Akhtar <i>et al.</i> [2]	5.28	5.27	5.76	5.41
Ours	0.74	0.69	1.39	0.46
Method	Img d7	Img h2	Img h3	Img f2
Huang <i>et al.</i> [11]	4.0	4.1	2.3	2.9
Kawakami <i>et al.</i> [12]	3.9	3.7	2.1	3.1
Nikon D700 response				
Akhtar <i>et al.</i> [2]	2.4	1.4	0.9	2.0
Ours	0.73	0.56	0.84	1.74
Our estimated response				
Akhtar <i>et al.</i> [2]	5.6	4.05	5.07	4.76
Ours	0.69	0.44	0.5	1.65

Table 3: Upsampling of NUS dataset by $32\times$ upsampling.

Method	Day 7	2500K 2	2500K 3	3000K 2	3000K 3
Akhtar <i>et al.</i> [2]	22.97	26.41	32.13	41.36	18.83
Ours	5.5	4.97	5.94	7.88	4.35

third examples was 2500K, and the forth and the fifth was 3000K.

Before using the dataset, we applied the preprocessing according to [19] to remove presenting biases in illumination. We normalized hyperspectrum with regard to illumination, and then integrated normalized hyperspectrum with given camera spectra response for RGB channel. After that, we applied white balance to get the corresponding RGB image. For further details of preprocessing and NUS dataset, please refer [19].

Our result outperformed Akhtar *et al.* [2] for all test examples as shown in the Table 3. In the NUS dataset, there is severe amount of noise in their hyperspectral images especially in low wavelength bands. The noise problem makes significant degradation in matrix factorization based approaches. In contrast, because our spatial upsampling stage includes denoising with the structure guided total variation regularization, our algorithm is robust against noise degradation.

4.4. Kawakami dataset

Our approach was tested on Kawakami dataset. The Kawakami dataset offers high resolution RGB images and low resolution hyperspectral images captured with their real world system. Thus, only qualitative comparisons are provided as there is no ground truth high resolution hyperspectral images. Since their dataset has spectrum of 415nm to 730nm sampled at every 5nm, and the given Kodak KAI-11002 sensor sensitivity has value from 400nm to 820nm for every 10nm, we applied spline interpolation to obtain

the value of the spectral response function at every 5nm. The scaling factors of this dataset varies between $6\times$ to $8\times$. Please refer Kawakami *et al.* [12] for acquisition process of their dataset.

Due to the very narrow bandwidth of the filters, the hyperspectral images in the Kawakami dataset have the most noise degradation and also with some blurs. Figure 4 shows the upsampled results and comparisons. As shown in the figure, there were severe block artifacts in the result from Akhtar *et al.* [2]. This came from their instability to handle noisy spectral components. Moreover, when severely under noise degradation, matrix factorization [2] sometimes failed to converge. This happened for the sheep and dinosaur cases (in the supplementary material), even with their proposing parameter in source codes. We observed that some results in the Kawakami dataset contains artifacts at boundary, these artifacts are caused by misalignment between high resolution RGB images and low resolution hyperspectral images.

4.5. Effect of Different Setting

Finally, we have also tested effects of different settings of our own algorithm using the CAVE dataset: Pan-guided, Spectral Substitution Only (hyperspectral and RGB), Spectral Substitution Only (RGB only), and Spatial Upsampling only. We included the bicubic interpolation for base-line comparison. Experiment results and some examples are shown in Table 4.

Pan guided We regarded the Y-channel as PAN image for this testing. In this testing, RGB images are not used. With only the PAN image, we could get high resolution structure, but spectral distortion was very severe. This is because spectral information in the PAN image is very limited. In addition, since Y-channel has more weighting in the middle wavelength band, *e.g.* G band, this makes spectral component in other bands to strongly follow spectral feature in the middle bands.















Spectral Substitution Only(hyperspectral and RGB) & Spatial Upsampling Only We skipped spatial upsampling part to verify the effect of spectral substitution, and vice versa. When using the spectral substitution only, its results were very similar to its original bicubic interpolation input. Since spectral substitution step only considers joint spectral distribution of given input, its estimated sparse coefficients was highly biased by the quality of intermediate results. The spectral substitution step was effective when accurate estimation was given. From this experiment, it shows that the spatial upsampling and the spectral substitution help each other in a complementary manner.

Spectral Substitution with RGB only We applied spectral substitution with only RGB input and without multi-scale. The sparse coefficient is estimated by solving Equation (5) with only the RGB inputs. This concept is similar to previ-

Table 4: Comparison of proposing approach with different settings.(32×). And the qualitative examples from Flowers.

Method	Balloons	Beads	Sponges	Oil paintings
Bicubic	13.07	32.22	19.5	14.22
Pan-guided	13.13	30.59	22.47	15.0
Spectral-Only	13.14	32.33	19.45	14.29
RGB-Only	20.53	35.13	28.07	18.14
Spatial-Only	1.79	7.07	2.99	3.74
Ours	1.64	6.92	2.89	3.55

Method	Flowers	CD	Fake and real peppers	photo and face
Bicubic	14.27	13.14	13.94	14.01
Pan-guided	13.41	11.63	14.58	14.13
Spectral-Only	14.34	13.24	14.01	14.03
RGB-Only	21.22	18.63	20.58	14.62
Spatial-Only	2.72	3.71	2.01	2.53
Ours	2.61	3.48	1.96	2.45

RGB	Y-channel	490nm	Input	
				
Bicubic	Pan-guide	Spectral	RGB	Spatial
				
				
9.96	10.34	10.03	17.54	2.31
RMSE				

ous matrix factorization works that we predict hyperspectral image only with high resolution RGB input and joint dictionary in single step. In the experiment, the worst result was shown. In our analysis, the high RMSE is caused by high ambiguity of hyperspectrum, if multiple objects with similar RGB colors presented in the same scene.

Parameters We performed additional experiments on the effect of segmentation parameters, i.e., number of clusters at each level, and the corresponding computation time. We tested on the Table 5, Flowers example (512x512 resolution). The implementation environment is Matlab 2013a, with Intel i7-4770 CPU @ 3.40 GHz, and 16 GB RAM. Increasing the number of segmentations reduces RMSE, but it also increase running time.

4.6. Spectra Accuracy

Figure 5 shows reconstructed spectrum of certain pixels in two of CAFE examples. The first example represents the same material with different RGB color, and the second example represents different materials with the same RGB color. As shown in the figure, matrix factorization method [2] has certain amount of distortion in recovered spectrum for the first case. Their method distorted the spec-

Table 5: Comparison for different segmentation parameter for Spectral substitution. Tested on × 32 for Table 1. Flowers. Bold text is our main setting for comparison with other State-of-the-arts.

	Segmentation	RMSE	time (.sec)
1	Spatial-Only	2.72	993
2	4-10-25-75-150	2.65	1710
3	5-13-33-100-200	2.62	1947
4	8-20-50-150-300	2.61	2090
5	12-30-75-225-450	2.58	2726
6	16-40-100-300-600	2.56	3051

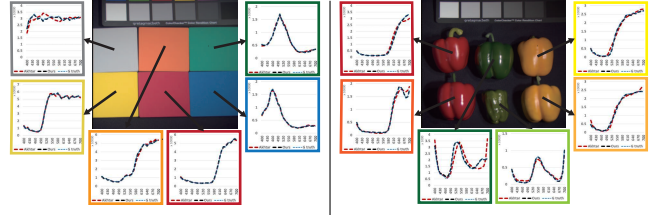


Figure 5: Comparison of spectra: Sponges(left) and Fake and real peppers(right).

tra more severely for the second case because of the limitation of global assumption. In both cases, our local sparse coding recovered spectrum very closely to the ground truth.

5. Conclusion and Discussion

In this paper, we have presented an algorithm for hyperspectral image upsampling by approaching the problem in both spatial and spectral aspects. From the trained HR-LR exemplars, we directly produce high resolution hyperspectral images from its low resolution inputs with RGB guiding. The upsampled results are further refined in spectral aspect by utilizing spectrum substitution. Extensive experiments show that our algorithm has outperformed state-of-the-art approaches. In addition, we have also performed a self-evaluation on our algorithm to better understand the effect of different setting, as well as effects of each step.

Contrast to the dataset in our experiment, hyperspectral signals in remote-sensing domain have wider wavelength range beyond visible RGB, the near and shortwave infrared(IR) [7]. With presence of near and short-wave IR, our locality inspired work will have more space to handle such highly nonlinear cases compared to previous state-of-the-arts, since our work utilizes high-resolution hyperspectrum information additional to traditionally used low-resolution counterpart and RGB. However, it will still have inevitably spectral distortions coming from insufficient mixing information between RGB and shortwave IR. Our next step is to exploit the possibility to apply our work to remote-sensing data where the HR PAN or the HR RGB images do not cover the entire range of hyperspectral inputs.

References

- [1] B. Aiazzi, S. Baronti, and M. Selva. Improving component substitution pansharpening through multivariate regression of ms+ pan data. *Geoscience and Remote Sensing, IEEE Transactions on*, 45(10):3230–3239, 2007. 5, 6
- [2] N. Akhtar, F. Shafait, and A. Mian. Sparse spatio-spectral representation for hyperspectral image super-resolution. *ECCV*, pages 63–78, 2014. 1, 2, 4, 5, 6, 7, 8
- [3] L. Alparone, L. Wald, J. Chanussot, C. Thomas, P. Gamba, and L. M. Bruce. Comparison of pansharpening algorithms: Outcome of the 2006 grs-s data-fusion contest. *Geoscience and Remote Sensing, IEEE Transactions on*, 45(10):3012–3021, 2007. 2
- [4] W. Caper, T. Lillesand, and R. Kiefer. The use of intensity-hue-saturation transforms for merging spot panchromatic and multispectral image data. *Photogrammetric Engineering and Remote Sensing*, 56(4):459–457, 1990. 2
- [5] A. Chakrabarti and T. Zickler. Statistics of Real-World Hyperspectral Images. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 193–200, 2011. 4
- [6] H. Chang, D. Y. Yeung, and Y. Xiong. Super-resolution through neighbor embedding. In *In Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, volume 1, pages I–I. IEEE, June 2004. 3
- [7] C. Chisense, J. Engels, M. Hahn, and E. Glch. Pansharpening of hyperspectral images in urban areas. In *Proc. XXII Congr. of the Int. Society for Photogrammetry, Remote Sens.*, volume XXXIX-B7, August 2012. 8
- [8] T. Goldstein and S. Osher. The split bregman method for l1-regularized problems. *SIAM Journal on Imaging Sciences*, 2(2):323–343, 2009. 3
- [9] H. F. Grahn and P. Geladi. *Techniques and applications of hyperspectral image analysis*. Wiley, 2007. 1
- [10] K. He, J. Sun, and X. Tang. Guided image filtering. *IEEE Trans. on PAMI*, 35(6):1397–1409, 2013. 2
- [11] B. Huang, H. Song, H. Cui, J. Peng, and Z. Xu. Spatial and spectral image fusion using sparse matrix factorization. *Geoscience and Remote Sensing, IEEE Transactions on*, 52(3):1693–1704, 2014. 2, 5, 7
- [12] R. Kawakami, J. Wright, Y. W. Tai, Y. Matsushita, M. Ben-Ezra, and K. Ikeuchi. High-resolution hyperspectral imaging via matrix factorization. In *CVPR*, pages 2329–2336, June 2011. 1, 2, 3, 4, 5, 6, 7
- [13] S. J. Kim, F. Deng, and M. S. Brown. Visual enhancement of old documents with hyperspectral imaging. *Pattern Recognition*, 44(7):1461–1469, 2011. 1
- [14] J. Kopf, M. F. Cohen, D. Lischinski, and M. Uyttendaele. Joint bilateral upsampling. *ACM Trans. on Graph.*, 26(3):96, 2007. 2
- [15] C. A. Laben and B. V. Brower. Process for enhancing the spatial resolution of multispectral imagery using pansharpening, US Patent 6011875, 2000. 2
- [16] H. Lee, A. Battle, R. Raina, and A. Y. Ng. Efficient sparse coding algorithms. *NIPS*, 19:801, 2007. 1, 4
- [17] Z. Li, X. M. Wu, and S. F. Chang. Segmentation using super-pixels: A bipartite graph partitioning approach. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 789–796. IEEE, 2012. 4
- [18] C. A. Micchelli, L. Shen, and Y. Xu. Proximity algorithms for image models: denoising. *Inverse Problem*, 27(4):045009, 2011. 3
- [19] R. M. Nguyen, D. K. Prasad, and M. S. Brown. Training-based spectral reconstruction from a single rgb image. *ECCV*, pages 186–201, 2014. 4, 7
- [20] J. Nunez, X. Otazu, O. Fors, A. Prades, V. Pala, and R. Arbiol. Multiresolution-based image fusion with additive wavelet decomposition. *IEEE Trans. on Geoscience and Remote Sensing*, 37(3):1204–1211, 1999. 2
- [21] M. Pegah, R. Anand, and G. Paul. Superpixel estimation for hyperspectral imagery. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 287–292, June 2014. 1
- [22] V. Shah, N. Younan, and R. King. An efficient pansharpening method via a combined adaptive pca approach and contourlets. *IEEE Trans. on Geoscience and Remote Sensing*, 46(5):1323–1335, 2008. 2
- [23] A. Vedaldi and B. Fulkerson. VLFeat: An open and portable library of computer vision algorithms. <http://www.vlfeat.org/>, 2008. 2
- [24] E. Wycoff, T. H. Chan, K. Jia, W. K. Ma, and Y. 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, May 2013. 2
- [25] C. Y. Yang and M. H. Yang. Fast direct super-resolution by simple functions. In *ICCV*, pages 561–568, December 2013. 2, 3
- [26] J. Yang, Z. Wang, Z. Lin, S. Cohen, and T. Huang. Coupled dictionary training for image super-resolution. *IEEE Trans. on Image Processing*, 21(8):3467–3478, 2012. 4
- [27] F. Yasuma, T. Mitsunaga, D. Iso, and S. Nayar. Generalized assorted pixel camera: Post-capture control of resolution, dynamic range and spectrum. Technical report, November 2008. 2, 4
- [28] N. Yokoya, T. Yairi, and A. Iwasaki. Coupled non-negative matrix factorization (cnmf) for hyperspectral and multispectral data fusion: Application to pasture classification. In *Geoscience and Remote Sensing Symposium (IGARSS), 2011 IEEE International*, pages 1779–1782. IEEE, 2011. 2
- [29] Y. Zhao, J. Yang, Q. Zhang, L. Song, Y. Cheng, and Q. Pan. Hyperspectral imagery super-resolution by sparse representation and spectral regularization. *EURASIP Journal on Advances in Signal Processing*, 1:1–10, 2011. 2
- [30] Y. Zongze, S. Hao, J. Kefeng, and Z. Huanxin. Hyperspectral image classification using discriminative dictionary learning. *IOP Conference Series Earth and Environmental Science*, 17(1):012222, 2014. 1