陕西省语音与图像信息处理重点实验室

多维信息感知与学习实验室

Multi-dimensional Information Sensing & Learning Lab

A Multi-scale CNN for Single Image Spectral Super-resolution

基于多尺度卷积神经网络的单图光谱超分辨

Yiqi Yan

Supervisor: Dr. Wei Wei

July 7, 2018

答辩人：闫奕岐

指导老师：魏巍

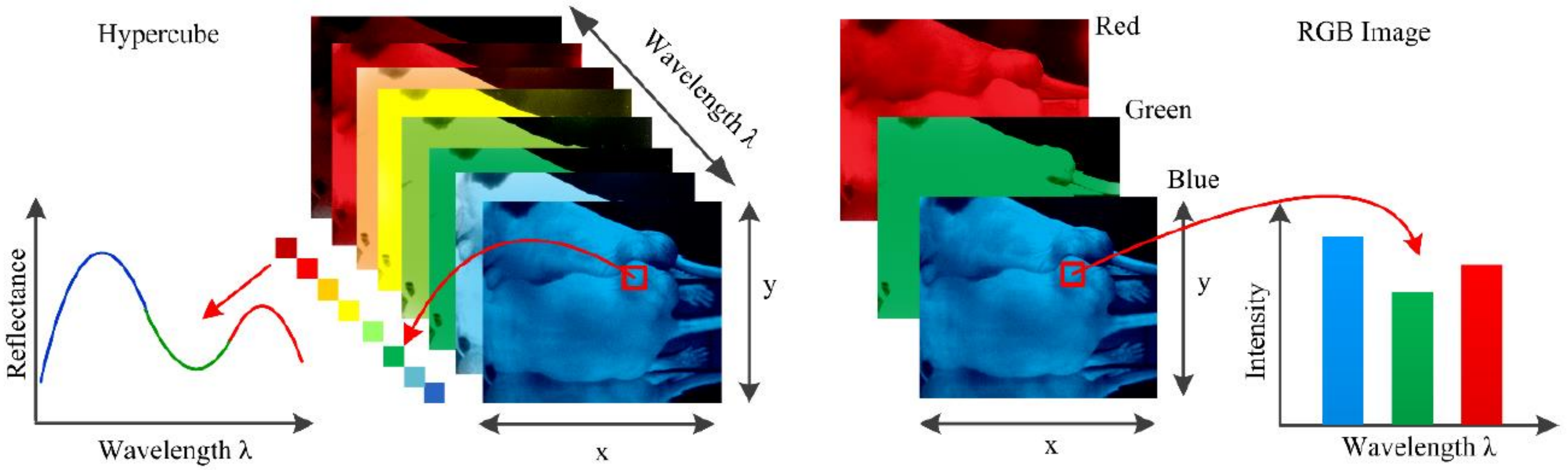
2018年6月7日

Contents

- Background
- Proposed method
- Comparison methods
- Implementation details
- Experimental results

- Background
- Proposed Method
- Comparison Methods
- Implementation Details
- Experimental Results

Hyperspectral Imaging



- narrow wavelength interval (*e.g.* 10nm)
- abundant spectral information

Hyperspectral Imaging: Application

Object Tracking

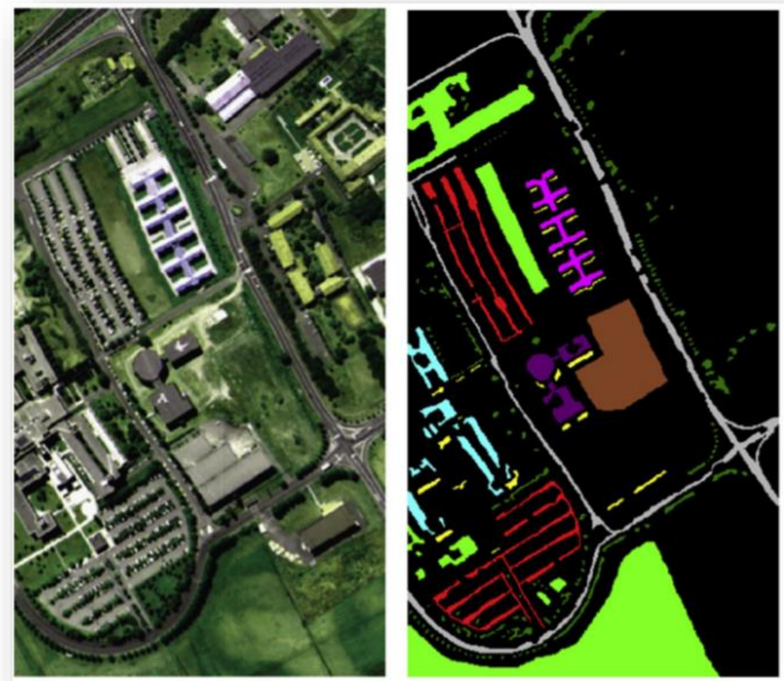
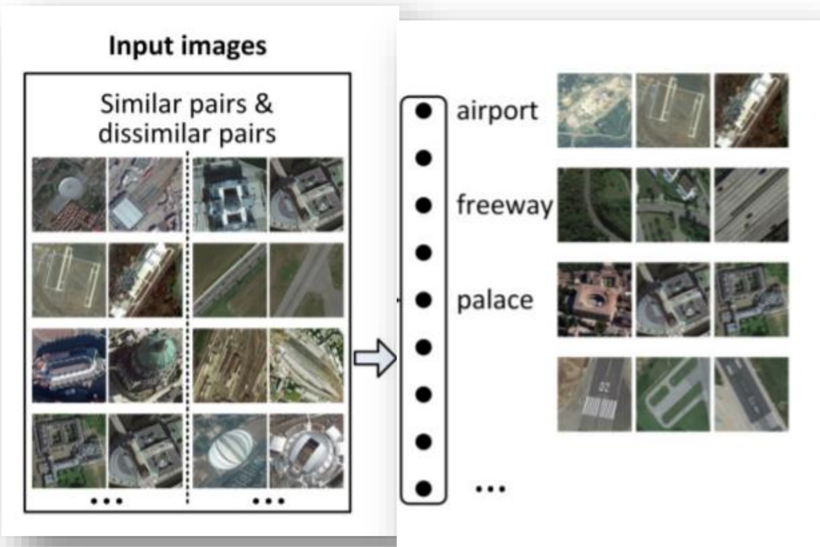


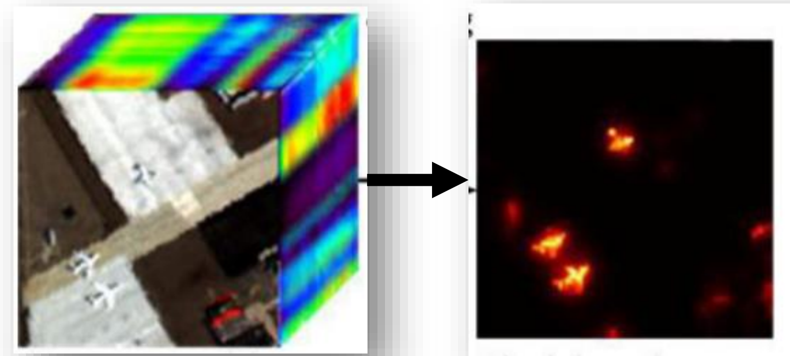
Image Segmentation

- H.V. Nguyen et al. Tracking via object reflectance using a hyperspectral video camera.
- Y. Tarabalka et al. Segmentation and classification of hyperspectral images using watershed transformation. Pattern Recognition, 43(7):2367–2379, 2010.

Hyperspectral Imaging: Application



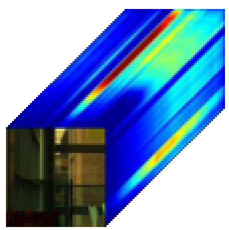
Scene Classification



Anomaly Detection

- G. Cheng et al. When deep learning meets metric learning: Remote sensing image scene classification via learning discriminative CNNs. IEEE Transactions on Geoscience and Remote Sensing, 2018.
- X. Kang et al. Hyperspectral anomaly detection with attribute and edge-preserving filters. IEEE Transactions on Geoscience and Remote Sensing, 55(10):5600–5611, 2017.

Hyperspectral Imaging: Practical Problem



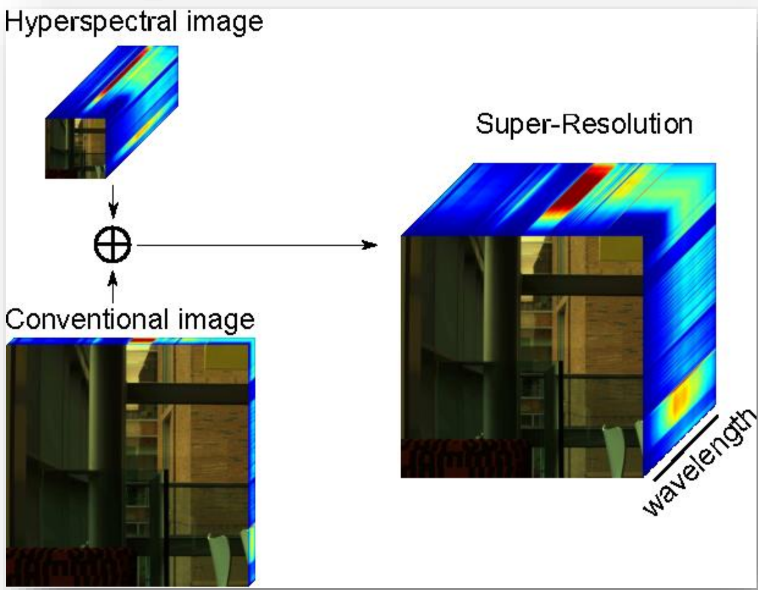
V.S

It is hard to directly acquire “ fully high-resolution” image

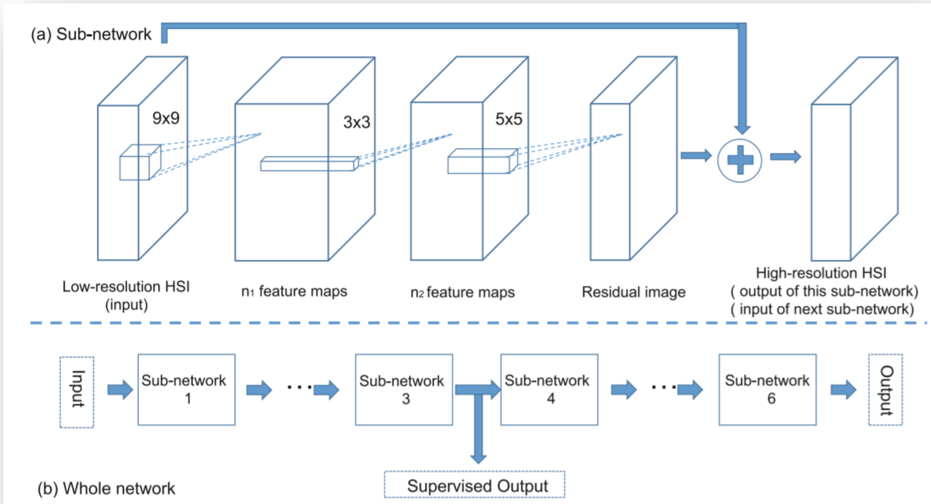
- Hyperspectral image: low spatial resolution
- Conventional image: low spectral resolution

Solution: Super-resolution Methods

—— *spatial domain*



Fusion-based Method



Deep Learning

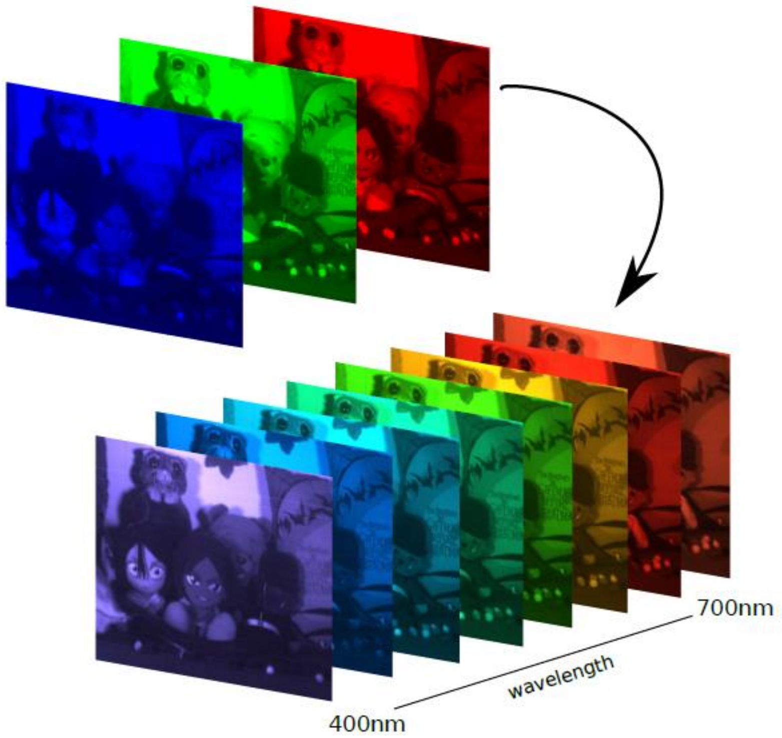
- C. Lanaras et al. Hyperspectral super-resolution by coupled spectral unmixing. CVPR 2015
- Wang et al. Deep residual convolutional neural network for hyperspectral image super-resolution. ICIGP 2017.

Solution: Super-resolution Methods
—— *spatial domain*

- High cost: still need hyperspectral sensors
- (Fusion-based) need well registered hyper-RGB image pair

Spatial super-resolution is still not practical in reality!

Solution: Super-resolution Methods
—— *spectral domain*



- Low cost: only RGB sensor is needed
- Single image: no need for extra data in addition to RGB images
- Our work focuses on spectral super-resolution

- Background
- Proposed Method
- Comparison Methods
- Implementation Details
- Experimental Results

Motivation

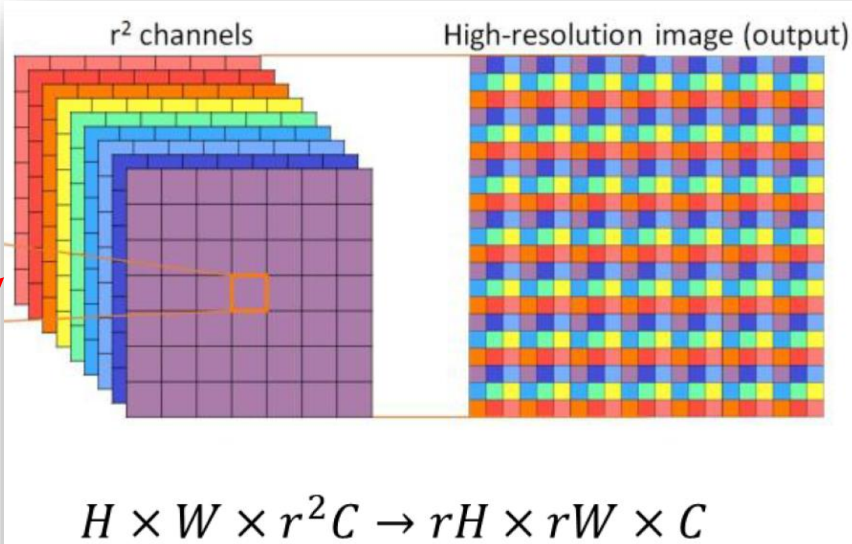
- Inherent correlation of natural images
- Local and non-local similarity
- Multi-scale information

Basic Building Blocks

Double Conv
3×3 convolution
Batch normalization
Leaky ReLU
Dropout
3×3 convolution
Batch normalization
Leaky ReLU
Dropout

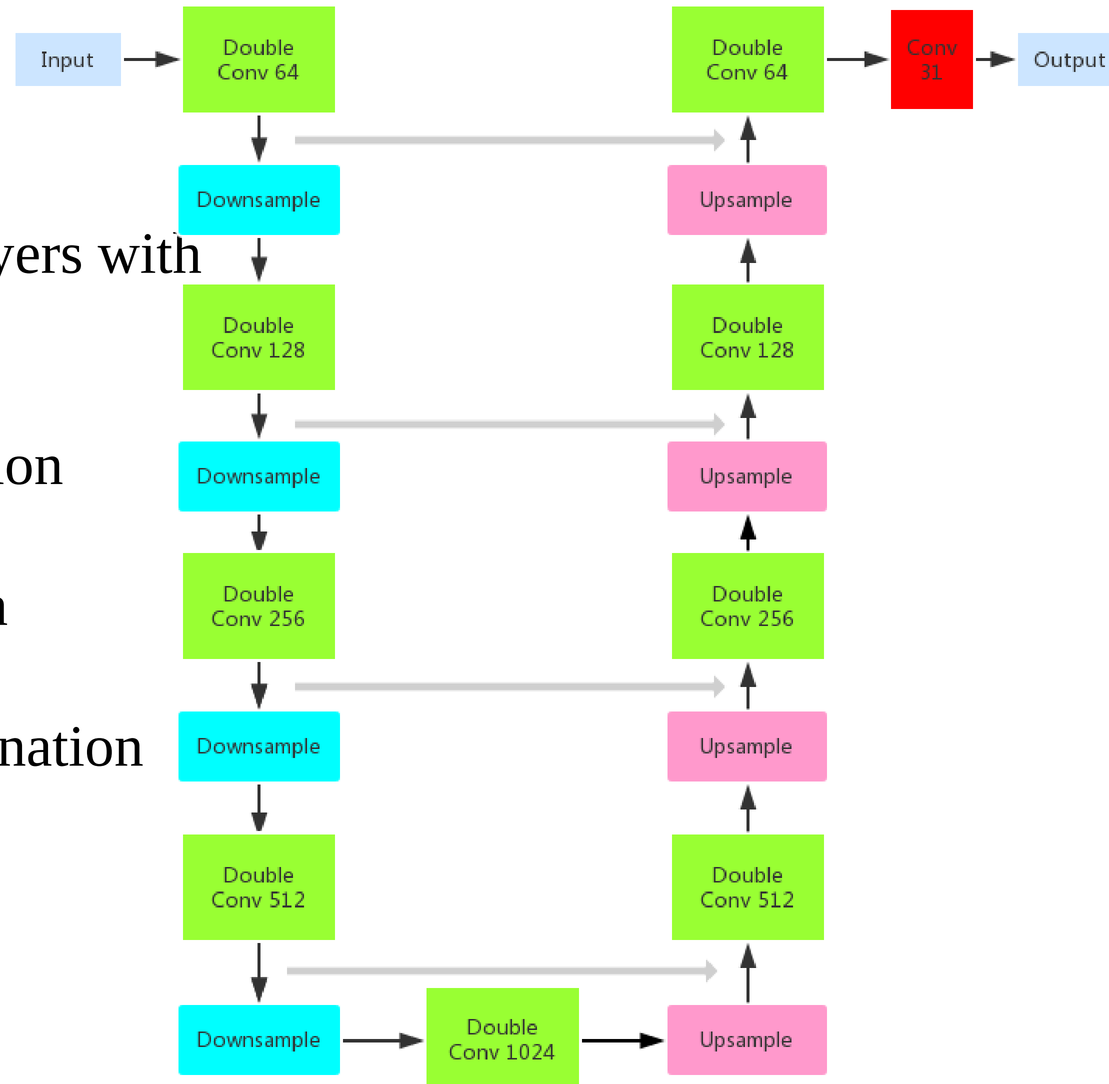
Downsample
2×2 max-pooling

Upsample
Pixel shuffle



- Conv: 3×3 convolution + batch normalization + leaky ReLU + dropout
- Downsample: regular max-pooling layer
- Upsample: pixel shuffle

Network Architecture



- “Conv m”: convolutional layers with an output of m feature maps
- green block: 3×3 convolution
- red block: 1×1 convolution
- gray arrows: feature concatenation

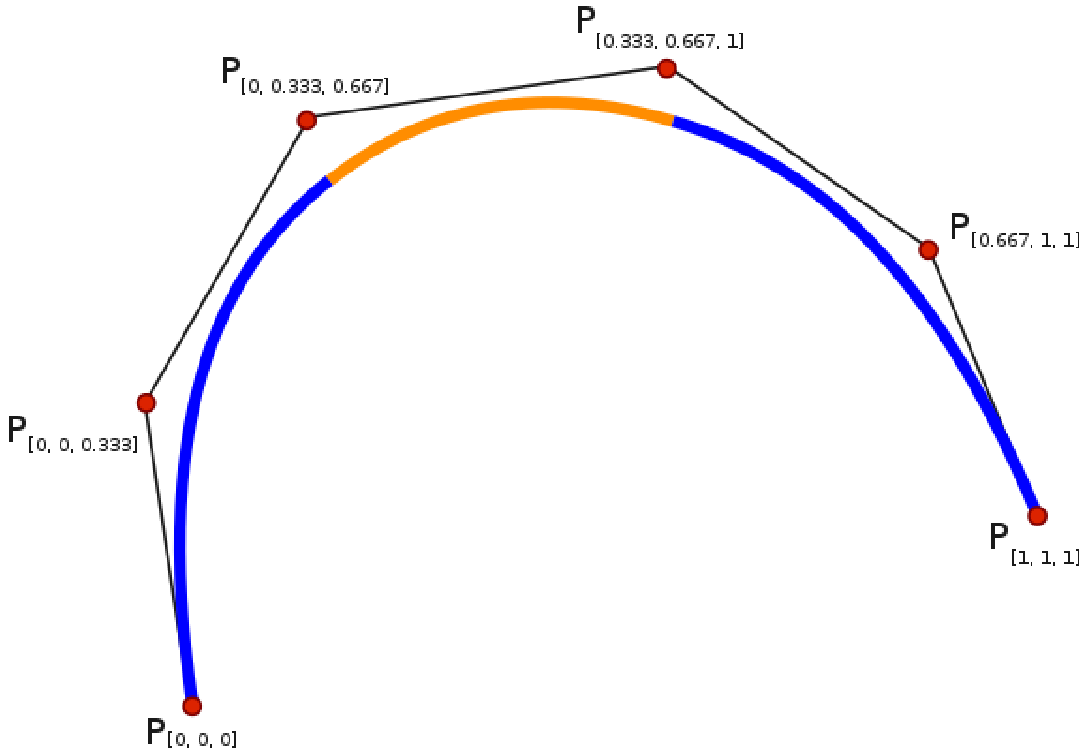
Intuition: encoder-decoder pattern

- Encoder
  - ✓ extracting features
  - ✓ increasing receptive field
  - ✓ non-local information
- Decoder
  - ✓ reconstructing spectra based on deep features
  - ✓ inducing multi-scale information by skip connections

- Background
- Proposed Method
- **Comparison Methods**
- Implementation Details
- Experimental Results

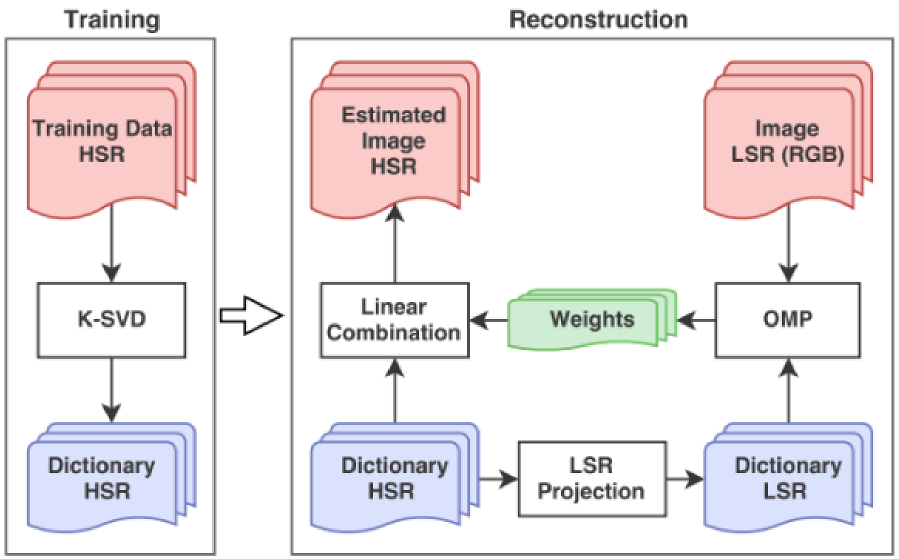
Baseline: Spline Interpolation

- a polynomial is assigned between each pair of data points
- the boundaries of polynomials are continuously differentiable
- provides small interpolation error despite the low degree of polynomials



Sparse Coding (Arad et al.)

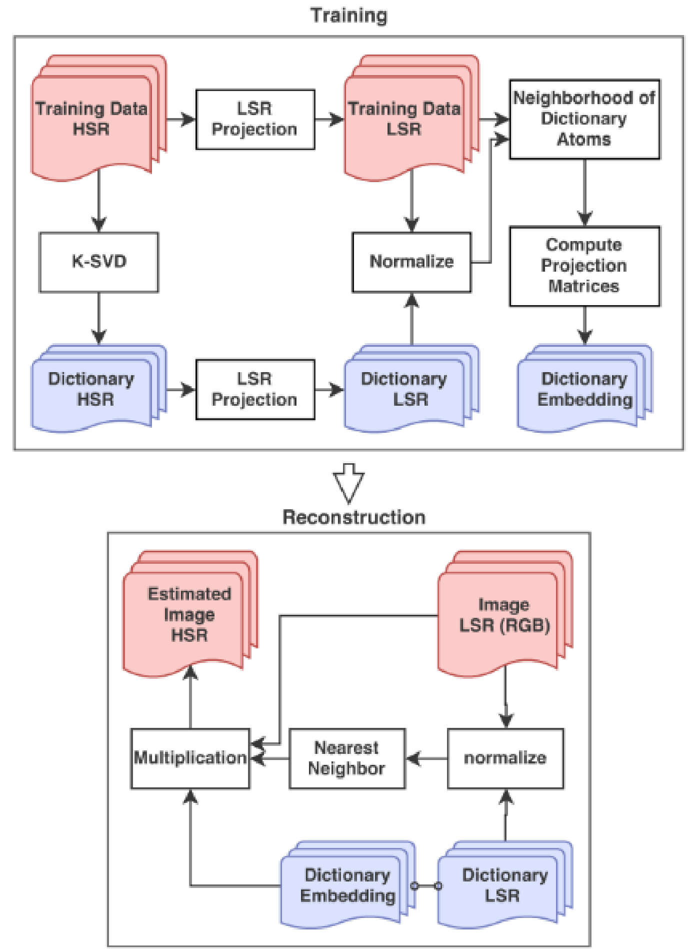
- Training: compute hyperspectral dictionary using *K-SVD*
- Reconstruction: compute sparse coefficients using *orthogonal matching pursuit (OMP)*



B. Arad et al. Sparse recovery of hyperspectral signal from natural RGB images. ECCV 2016.

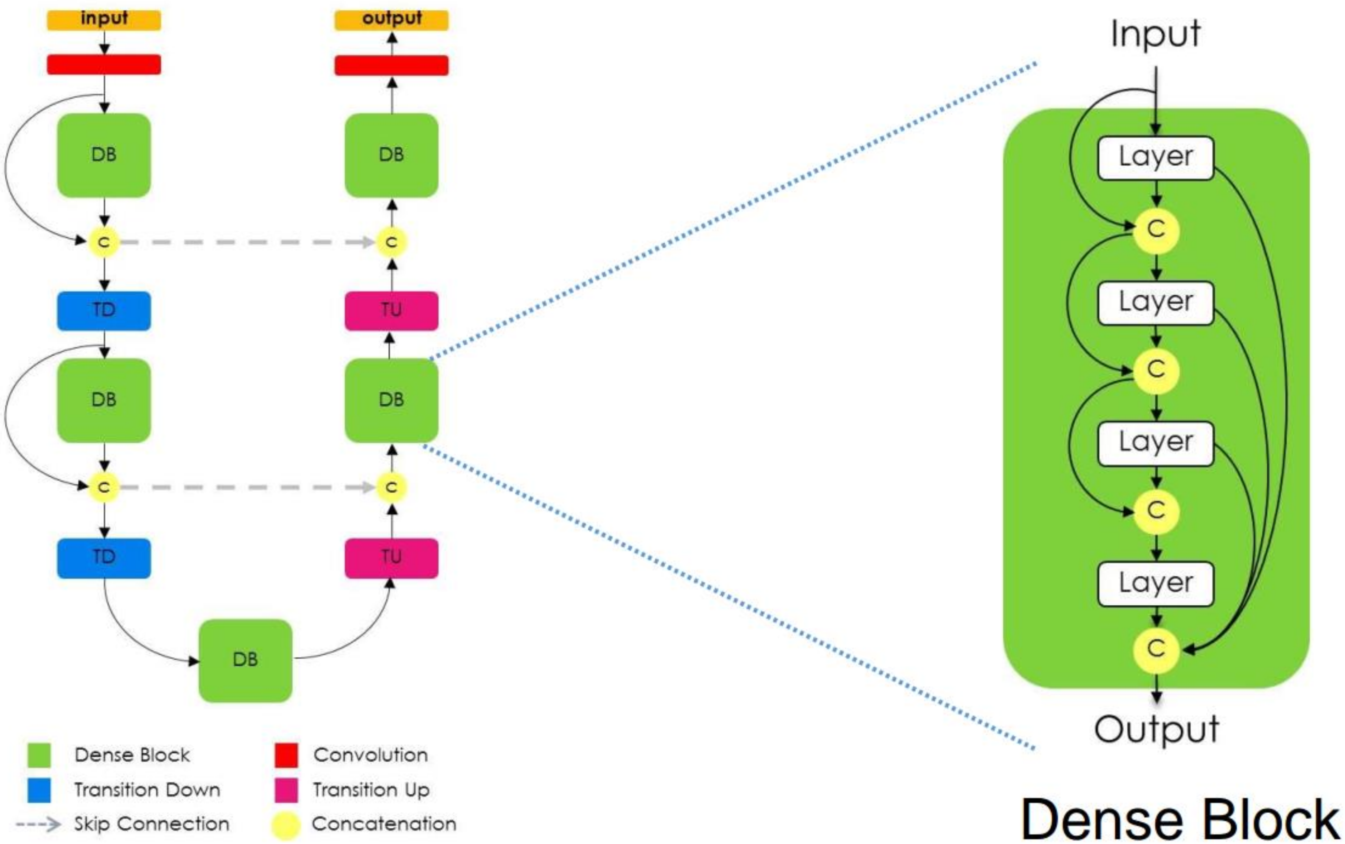
A+ Method

- Training: compute hyperspectral dictionary using *K-SVD*; compute sparse coefficients via *sparse least square problem*
- Offline compute and store the projection matrices
- Reconstruction: use the projection matrix to embed RGB samples into hyperspectra space



J. Aeschbacher et al. In defense of shallow learned spectral reconstruction from RGB images. CVPRW 2017

Deep Learning (Galliani et al.)



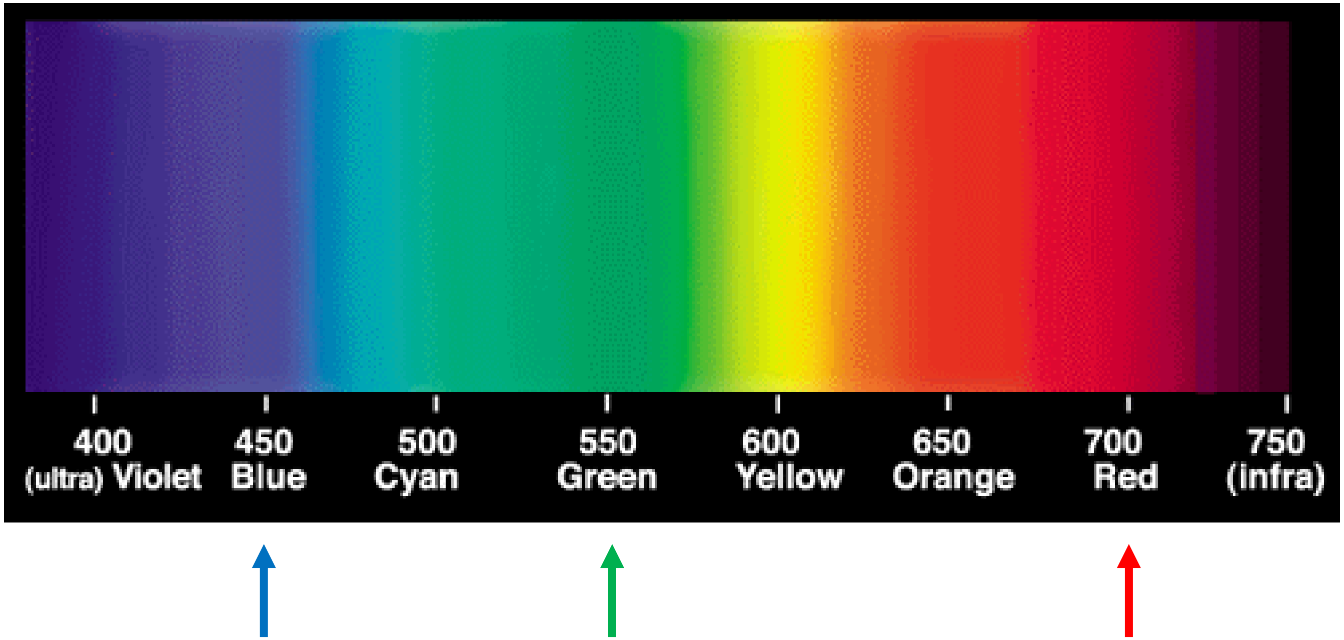
S. Galliani et al. Learned spectral super-resolution. CoRR, abs/1703.09470, 2017. http://arxiv.org/abs/1703.09470.

- Background
- Proposed Method
- Comparison Methods
- **Implementation Details**
- Experimental Results

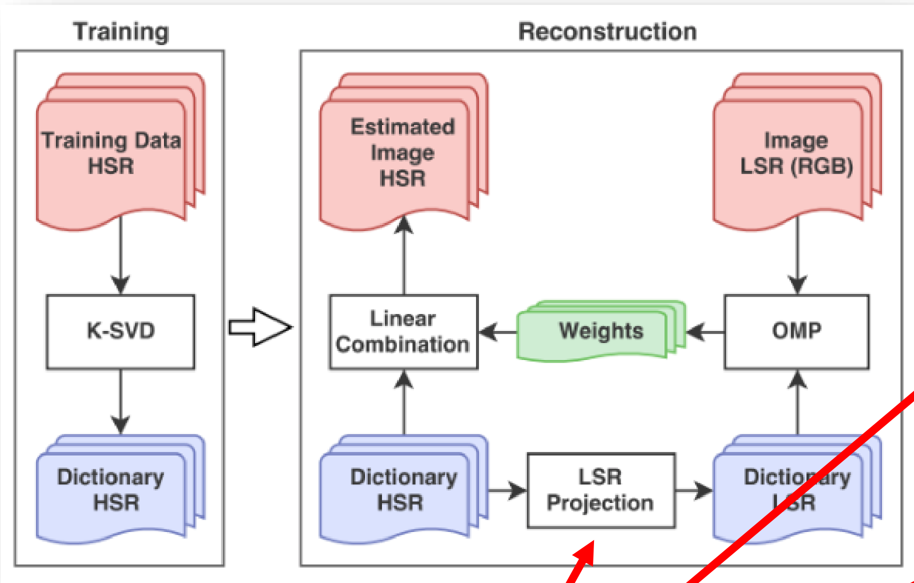
Spline Interpolation

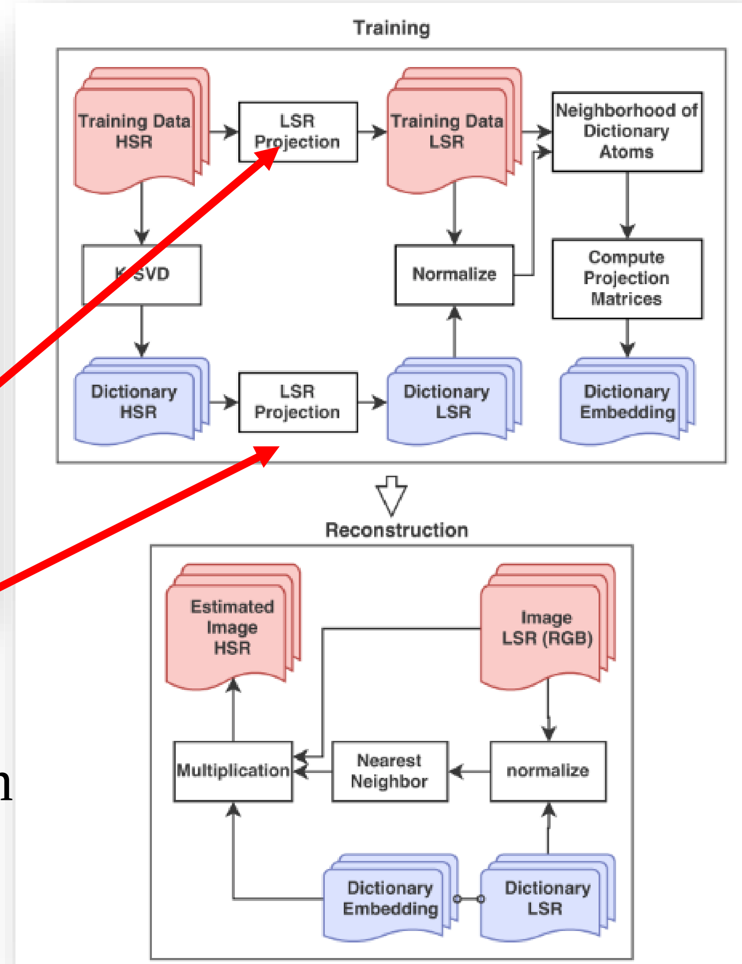
- data protocol: 31 bands; 400~700 *nm* with 10 *nm* interval
- MATLAB code

```
x = [31,16,6];
y = rgb;
xx = 1:31;
spectrum = spline(x,y,xx);
```



Sparse Coding Methods (Arad et al. & A+)





Fit the LSR projection matrix using trainin
data via regular *linear regression*

Deep Learning Methods (Galliani et al. & Ours)

- Hyper-parameters

	Galliani *et al.*	Ours
Dropout rate	0.5	0.2
Slope for leaky ReLU	0.2	0.2
Initial learning rate	2×10^{-3}	5×10^{-5}
Weight penalty	1×10^{-6}	1×10^{-6}
Weight initialization	HeUniform	HeNormal

- Optimizer: Adam

- Learning rate decay strategy

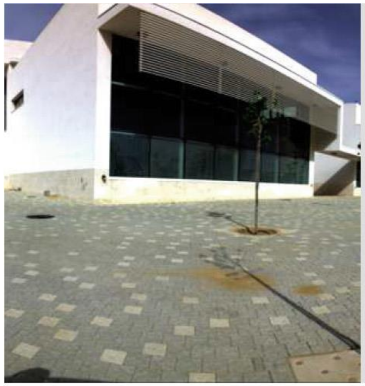
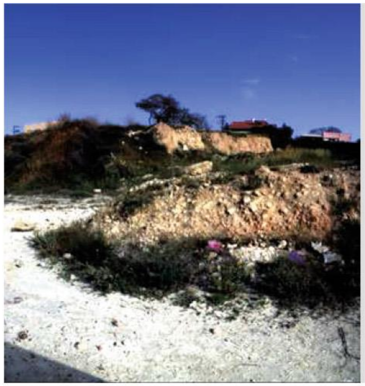
✓ Galliani et al.: 2×10^{-3} for 50 epochs + 2×10^{-4} for 50 epochs

✓ Ours: decayed by 0.93 every 10 epochs

Dataset: NTIRE2018

	number of images	size	bands	spectral band
NTIRE2018	256 training + 5 test	1392×1300	31	$400 \sim 700nm$
CAVE	32	512×512	31	$400 \sim 700nm$
HARVARD	50	1024×1024	31	$420 \sim 720nm$

NTIRE2018: latest & largest!

NTIRE 2018 challenge on spectral reconstruction from RGB images (CVPR 2018)
http://www.vision.ee.ethz.ch/ntire18/

Evaluation Metrics
—— *pixel-level reconstruction error*

- absolute root mean square error (RMSE)

$$RMSE_1 = \frac{1}{n}\sum_{i=1}^{n}\sqrt{\left(I_h^{(i)} - I_e^{(i)}\right)^2}$$

$$RMSE_2 = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(I_h^{(i)} - I_e^{(i)}\right)^2}$$

- relative root mean square error (rRMSE)

$$rRMSE_1 = \frac{1}{n}\sum_{i=1}^{n}\frac{\sqrt{\left(I_h^{(i)} - I_e^{(i)}\right)^2}}{I_h^{(i)}}$$

$$rRMSE_2 = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\frac{\left(I_h^{(i)} - I_e^{(i)}\right)^2}{\bar{I}_h^2}}$$

$I_h^{(i)}$, $I_e^{(i)}$: ith element of the real and estimated hyperspectral images

$\bar{I}_h$: the average of all elements in I_h

n: number of elements in an image

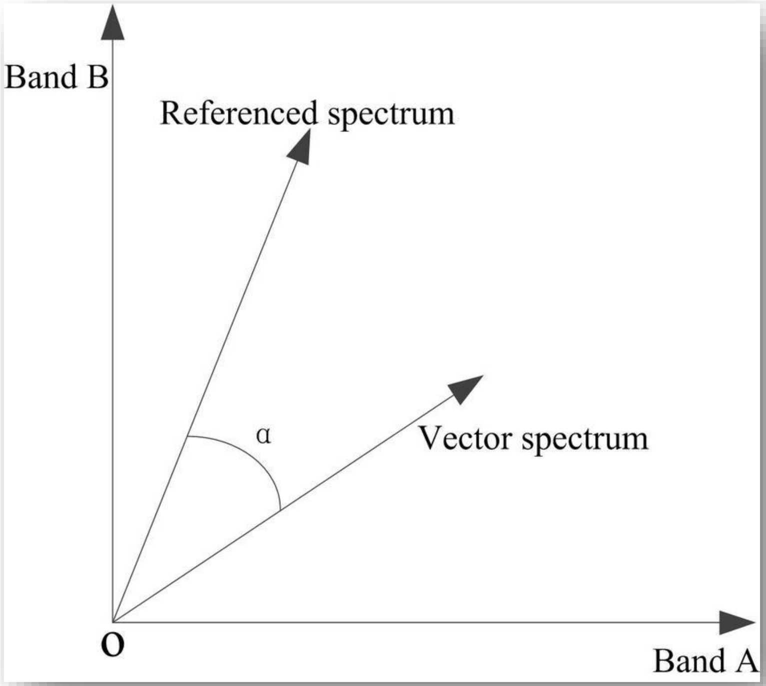
Evaluation Metrics
—— *spectral similarity*

- Spectral angle mapper

$$SAM = \frac{1}{m} cos^{-1} \left(\sum_{j=1}^{m} \frac{(\boldsymbol{p}_h^{(j)})^T \cdot \boldsymbol{p}_e^{(j)}}{\left\| \boldsymbol{p}_h^{(j)} \right\|_2 \cdot \left\| \boldsymbol{p}_e^{(j)} \right\|_2} \right)$$

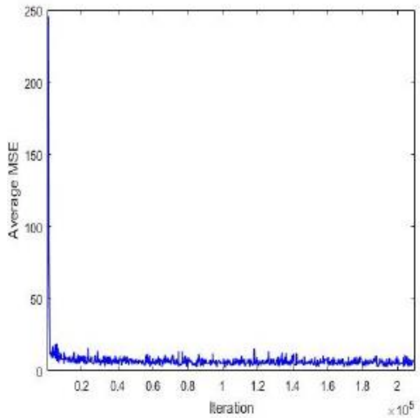
$p_h^{(j)}$, $p_e^{(j)}$: jth hyperspectral pixel in real and estimated hyperspectral images

m: number of pixels in an image

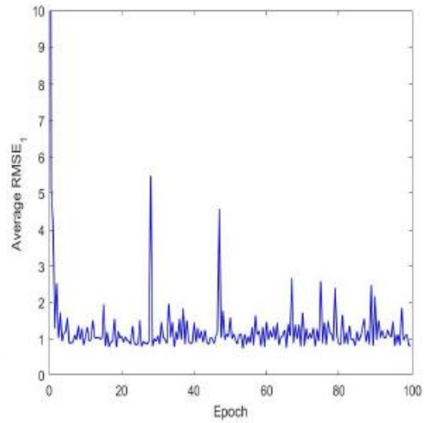


- Background
- Proposed Method
- Comparison Methods
- Implementation Details
- Experimental Results

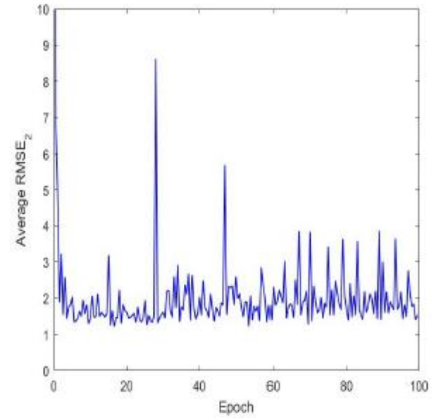
Convergence Analysis



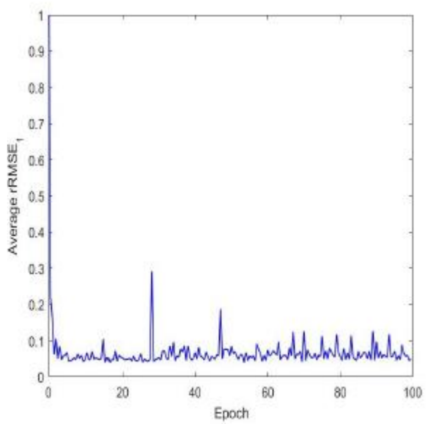
(a) Training curve



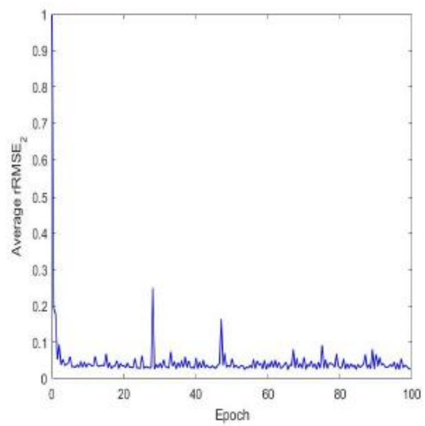
(b) $RMSE_1$ test curve



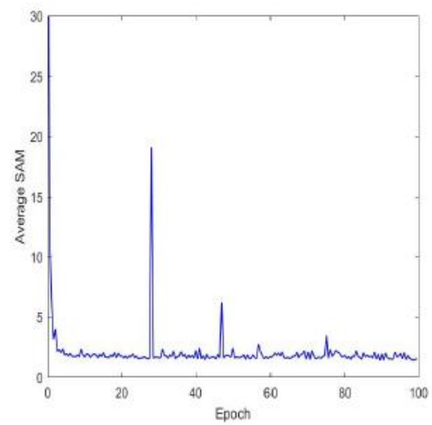
(c) $RMSE_2$ test curve



(d) $rRMSE_1$ test curve



(e) $rRMSE_2$ test curve



(f) SAM test curve

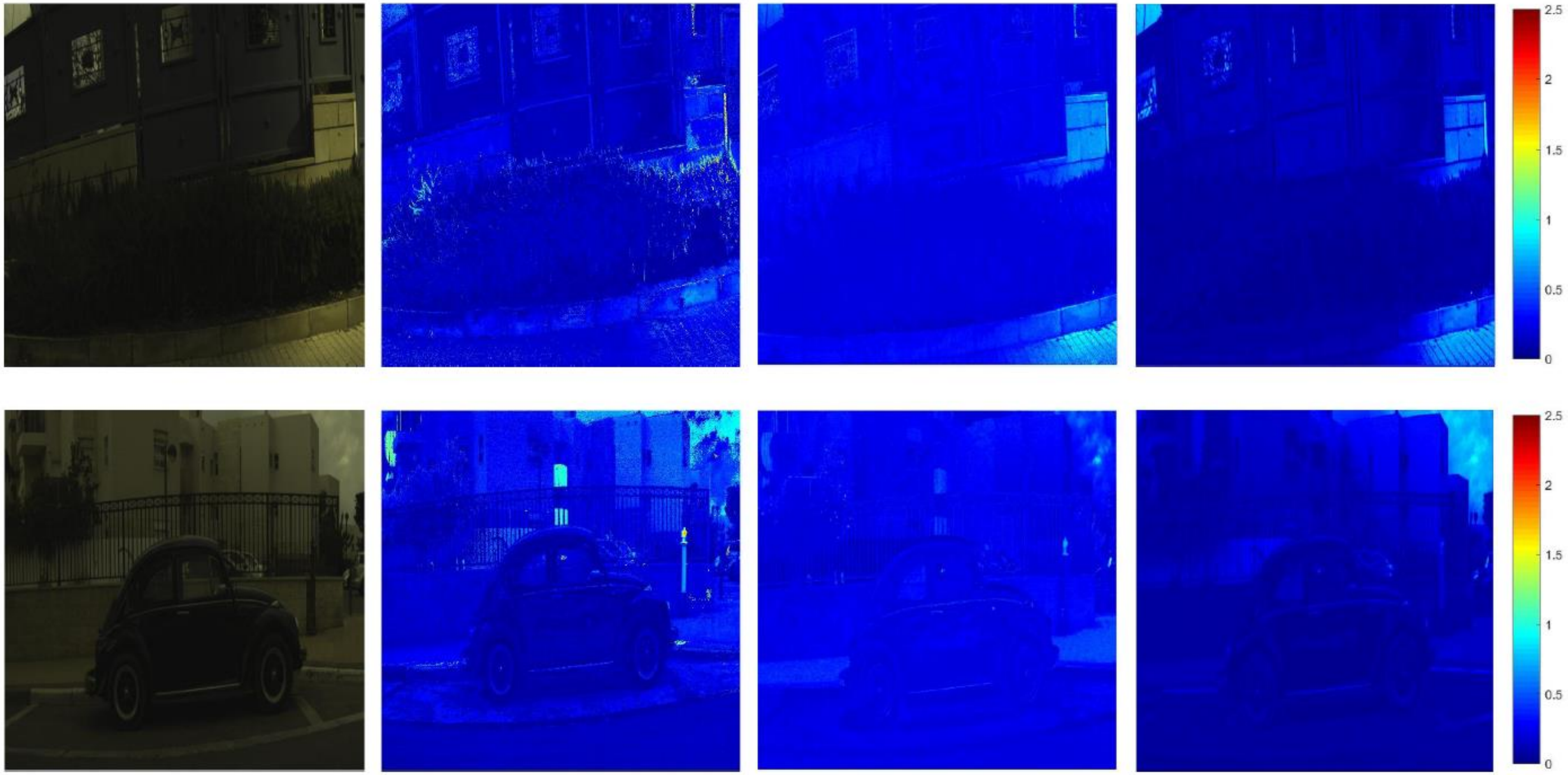
Quantitative Results

$RMSE_1$						
	BGU_00257	BGU_00259	BGU_00261	BGU_00263	BGU_00265	Average
Interpolation	1.8622	1.7198	2.8419	1.3657	1.9376	1.9454
Arad *et al.*	1.7930	1.4700	1.6592	1.8987	1.2559	1.6154
A+	1.3054	1.3572	1.3659	1.4884	0.9769	1.2988
Galliani *et al.*	0.7330	0.7922	0.8606	0.5786	**0.8276**	0.7584
Our	**0.6172**	**0.6865**	**0.9425**	**0.5049**	0.8375	**0.7177**
$RMSE_2$						
	BGU_00257	BGU_00259	BGU_00261	BGU_00263	BGU_00265	Average
Interpolation	3.0774	2.9878	4.1453	2.0874	3.9522	3.2500
Arad *et al.*	3.4618	2.3534	2.6236	2.5750	2.0169	2.6061
A+	2.1911	1.9572	1.9364	2.0488	1.3344	1.8936
Galliani *et al.*	1.2381	**1.2077**	**1.2577**	0.8381	**1.6810**	**1.2445**
Ours	**0.9768**	1.3417	1.6035	**0.7396**	1.7879	1.2899

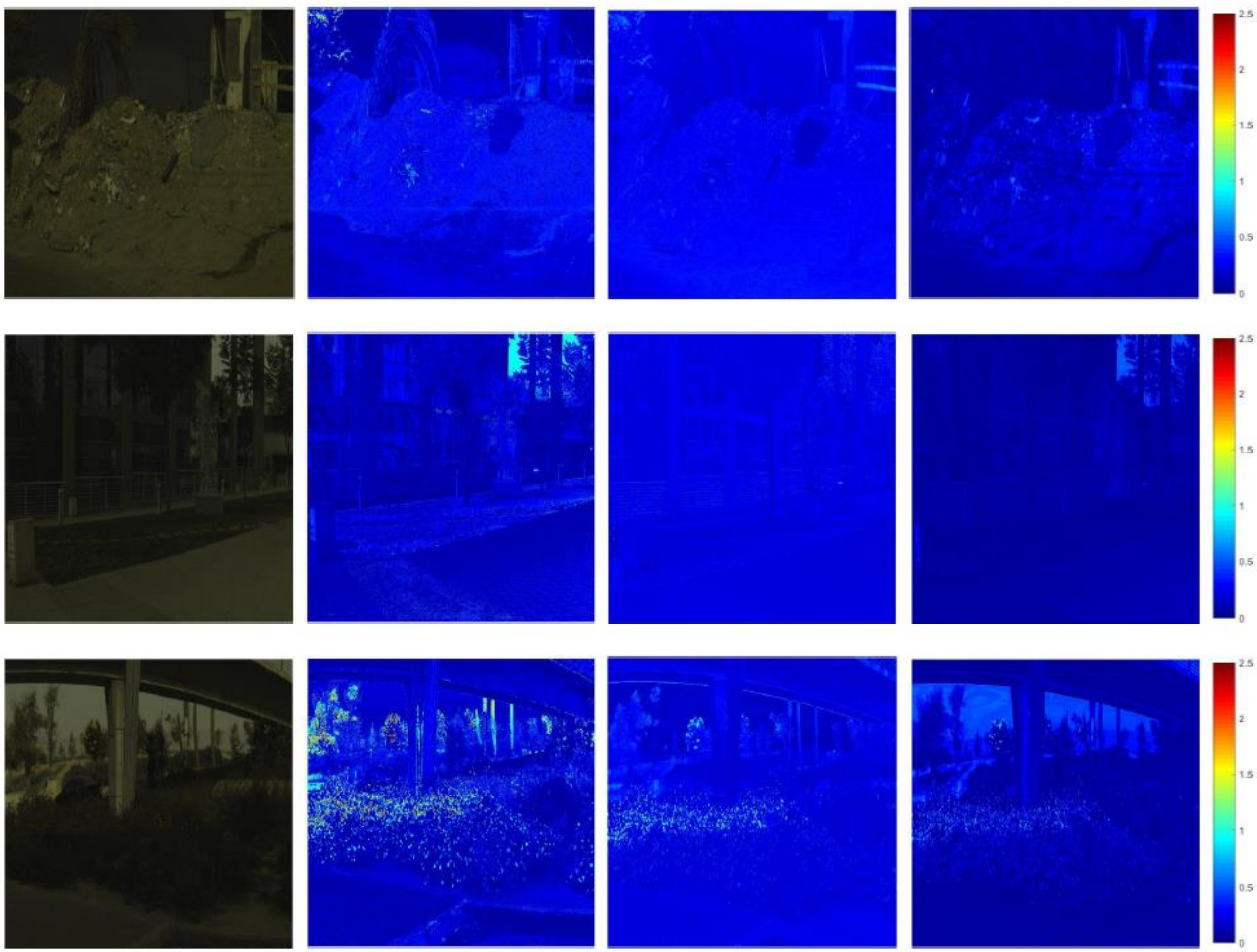
Quantitative Results

			$rRMSE_1$			
	BGU_00257	BGU_00259	BGU_00261	BGU_00263	BGU_00265	Average
Interpolation	0.0658	0.0518	0.0732	0.0530	0.0612	0.0610
Arad *et al.*	0.0807	0.0627	0.0624	0.0662	0.0560	0.0656
A+	0.0580	0.0589	0.0612	0.0614	0.0457	0.0570
Galliani *et al.*	0.0261	0.0268	0.0254	0.0237	0.0289	0.0262
Ours	**0.0235**	**0.0216**	**0.0230**	**0.0205**	**0.0278**	**0.0233**
			$rRMSE_2$			
	BGU_00257	BGU_00259	BGU_00261	BGU_00263	BGU_00265	Average
Interpolation	0.1058	0.0933	0.1103	0.0759	0.1338	0.1038
Arad *et al.*	0.1172	0.0809	0.0819	0.0685	0.0733	0.0844
A+	0.0580	0.0589	0.0612	0.0614	0.0457	0.0610
Galliani *et al.*	0.0453	**0.0372**	**0.0331**	0.0317	**0.0562**	**0.0407**
Ours	**0.0357**	0.0413	0.0422	**0.0280**	0.0598	0.0414
			SAM (degree)			
	BGU_00257	BGU_00259	BGU_00261	BGU_00263	BGU_00265	Average
Interpolation	3.9620	3.0304	4.2962	3.1900	3.9281	3.6813
Arad *et al.*	4.2667	3.7279	3.4726	3.3912	3.3699	3.6457
A+	3.2952	3.5812	3.2952	3.0256	3.2952	3.2985
Galliani *et al.*	1.4725	1.5013	**1.4802**	1.4844	**1.8229**	1.5523
Ours	**1.3305**	**1.2458**	1.7197	**1.1360**	1.9046	**1.4673**

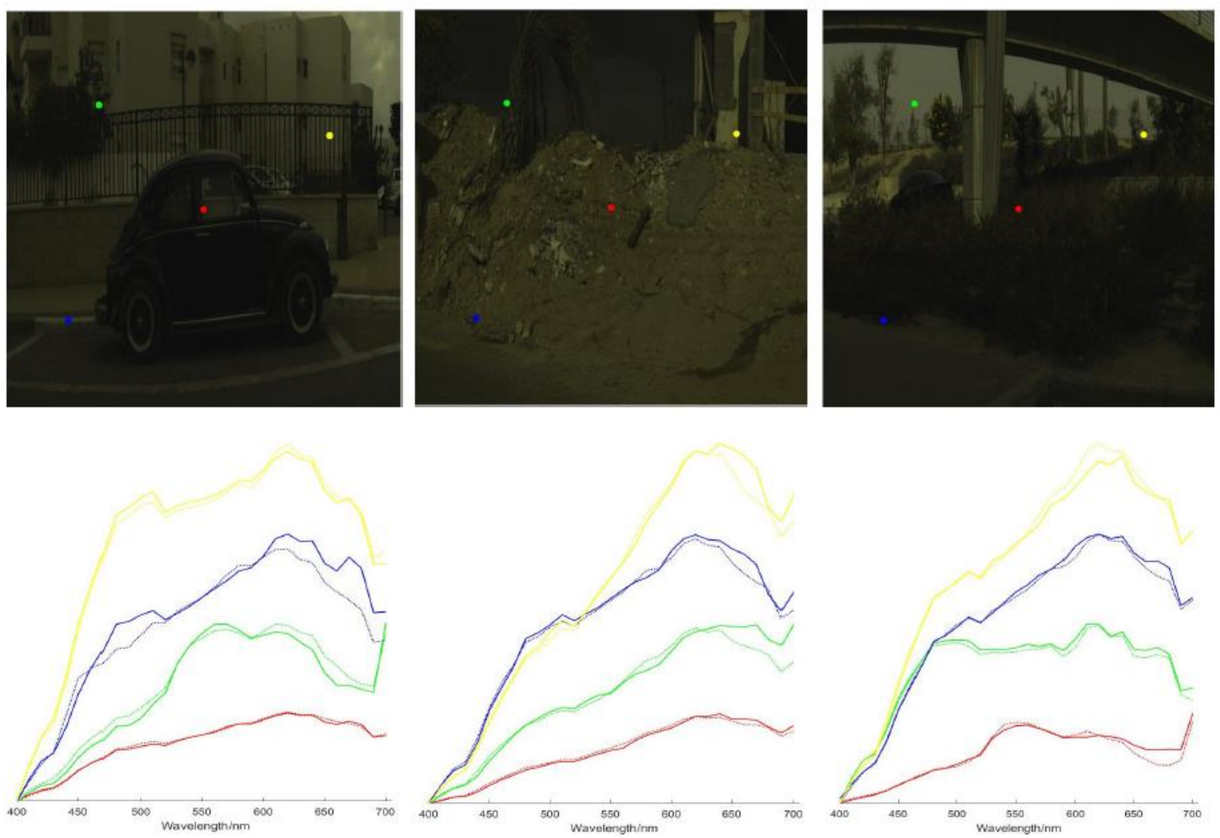
Visual Results



Visual Results



Visual Results

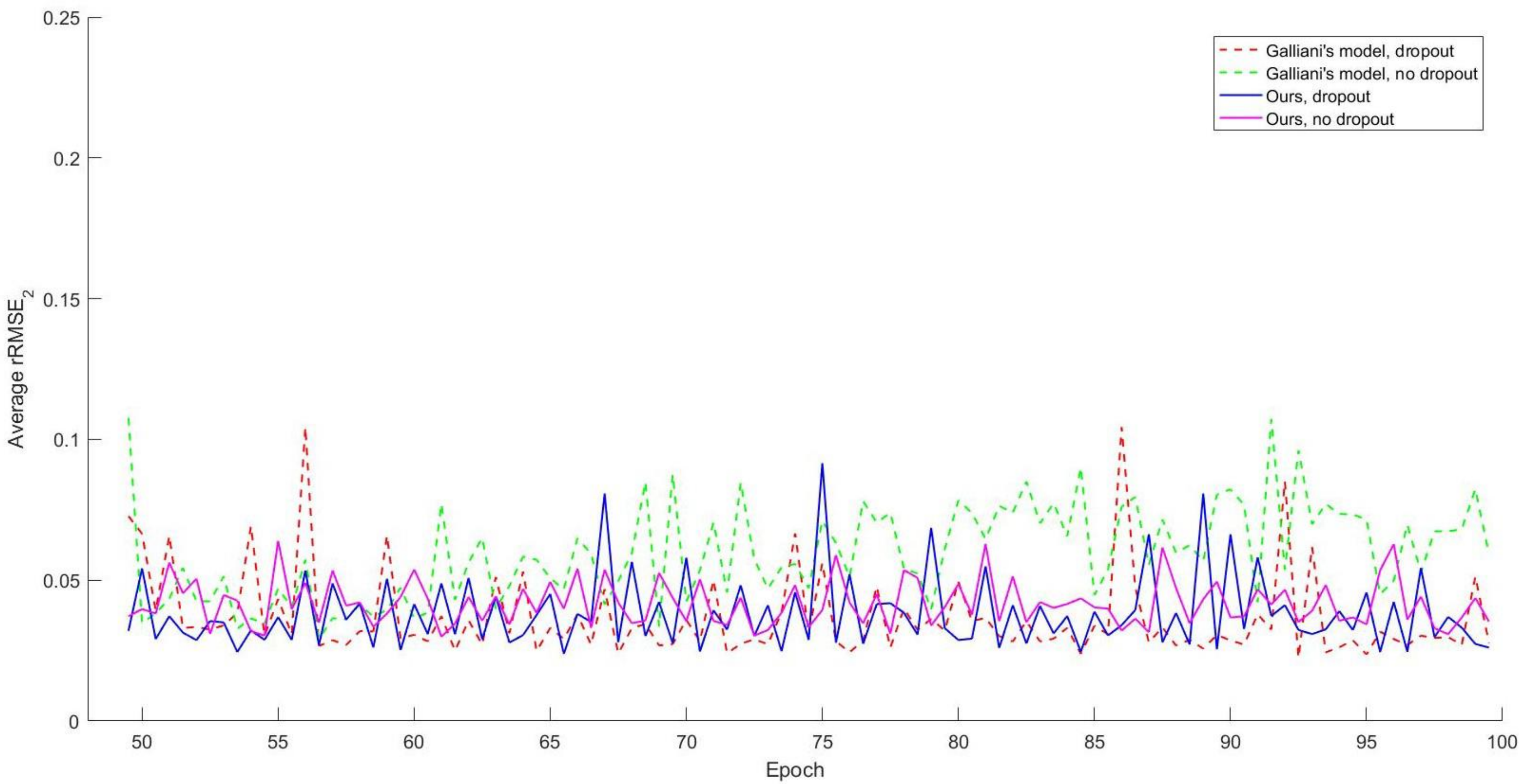


Sensitive Analysis

	Galliani *et al.*	Galliani *et al.* (no dropout)	Increment (%)	Ours	Ours (no dropout)	Increment (%)
$RMSE_1$	0.7584	1.6092	112.18	0.7177	1.0662	48.56
$RMSE_2$	1.2445	2.0492	64.66	1.2899	1.8168	40.85
$rRMSE_1$	0.0262	0.0617	135.50	0.0233	0.0320	37.34
$rRMSE_2$	0.0407	0.0673	65.36	0.0414	0.0593	43.24
SAM	1.5523	2.1358	37.59	1.4673	1.6206	10.45

our network is *more robust* and
less sensitive to hyper-parameters

Sensitive Analysis



Publication

Yiqi Yan, Lei Zhang, Wei Wei, Yanning Zhang, *Accurate Spectral Super-resolution from Single RGB Image Using Multi-scale CNN.* Submitted to Chinese Conference on Pattern Recognition and Computer Vision (PRCV) 2018