Hyperspectral Image Compression using Implicit Neural Representations

Shima Rezasoltani

Computer Science, Faculty of Science

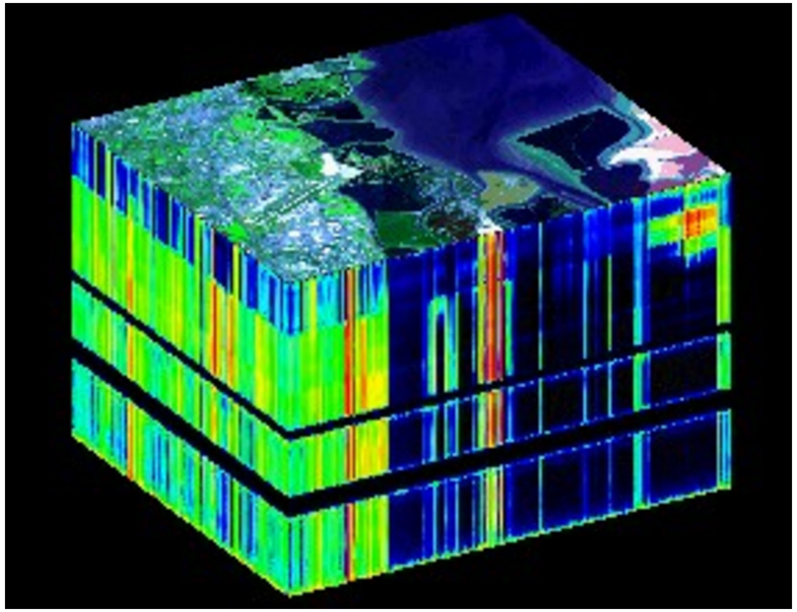
PhD Defense

OntarioTech UNIVERSITY

Visual Computing LAB

Hyperspectral Images

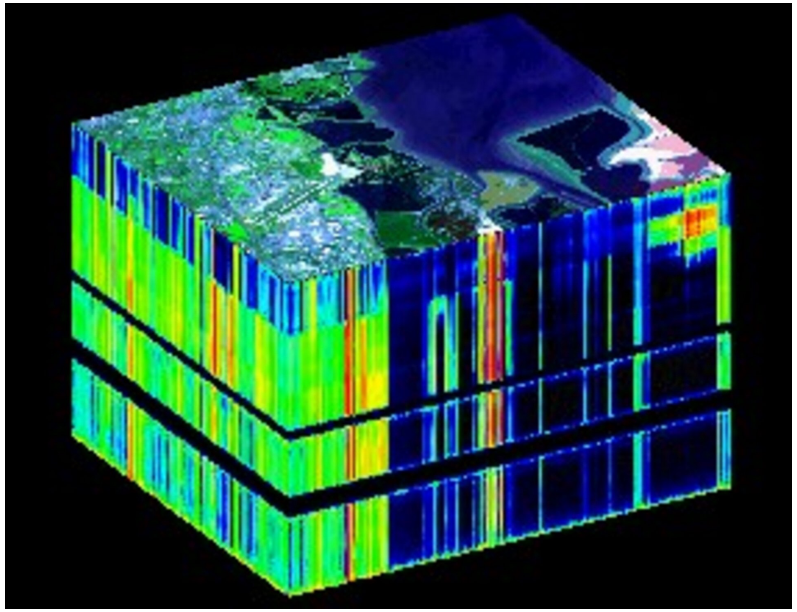
- A hyperspectral image is a data cube that captures spatial information across hundreds of contiguous spectral bands



Credit: European Space Agency, hyperspectral image “data cube.”

Hyperspectral Images

- A hyperspectral image is a data cube that captures spatial information across hundreds of contiguous spectral bands



Credit: European Space Agency, hyperspectral image “data cube.”

Enables detailed analysis of material properties based on their spectral signatures.

RGB vs. Hyperspectral Images

$$5712 \times 4284 \times 3 \approx 293 \text{ MB}$$

$$4192 \times 6708 \times 270 \approx 30 \text{ GB}$$

RGB vs. Hyperspectral Images

Hyperspectral images require orders of magnitude more storage space and bandwidth than those for RGB images.

Require efficient schemes for hyperspectral data compression in order to make hyperspectral images a practical choice for real world applications

$$5712 \times 4284 \times 3 \approx 293 \text{ MB}$$

Disk: 9 MB JPEG

$$4192 \times 6708 \times 270 \approx 30 \text{ GB}$$

Disk: 28 GB

Previous Work Highlights

- **Transform-based methods**
  - 3D Discrete Cosine Transform, Wavelength Transform, Tucker decomposition, compressed sensing approaches
- **Learning-based approaches**
  - Evolutionary approaches, Autoencoders
- **Dimensionality reduction**
  - Principle Component Analysis, band selection
- **Hyperspectral images as videos**
- **Region-aware schemes**

Previous Work Highlights

- **Transform-based methods**
  - 3D Discrete Cosine Transform, Wavelength Transform, Tucker decomposition, compressed sensing approaches
- **Learning-based approaches**
  - Evolutionary approaches, Autoencoders
- **Dimensionality reductio**
  - Principle Component An
- **Hyperspectral images as**
- **Region-aware schemes**

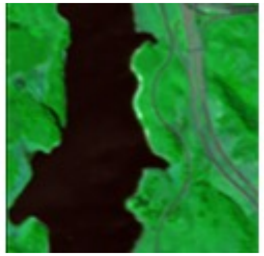
There is no clear winner, or generally agreed upon, scheme for compressing hyperspectral data.
There is no JPEG standard for hyperspectral images!

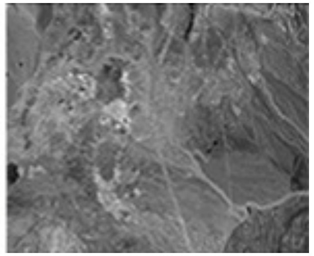
Thesis Focus

- Study and develop new methods for hyperspectral image compression

Emerging applications of hyperspectral images

Benchmarks



- **Indian Pines:** Commonly used for agricultural and vegetation analysis.
- **Jasper Ridge:** Features a mix of vegetation and urban features.
- **Pavia University:** High-resolution urban dataset.
- **Cuprite:** Primarily geological and mineralogical.

Widely used benchmarks for evaluating hyperspectral compression

Offer an opportunity to capture method’s performance in different application scenarios: urban, agricultural, minerology, etc.

Metrics

- **Peak Signal-to-Noise Ratio** (↑)
  - Quantifies the reconstruction quality by comparing the similarity between the original and compressed images using an interpretable logarithmic scale.
- **Structural Similarity Index Metric** (↑)
  - Evaluates perceived image quality by modeling structural information, luminance, and contrast, aligning more closely with human visual perception.
- **Bits per pixel per band**
  - Captures the level of compression achieved by a model. (↓)

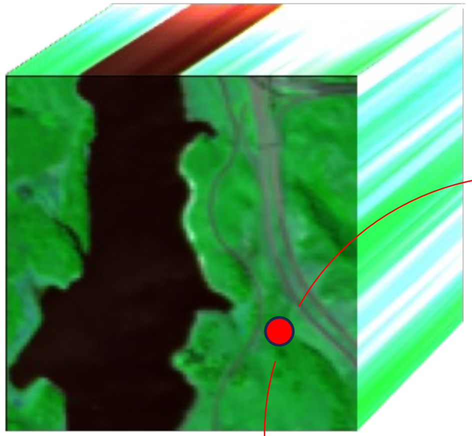
Implicit Neural Representations (INRs)

- **Idea:** Represent hyperspectral images as *implicit neural representations* [Dupont *et al.*, 2021]

Networks that map a (pixel) location to (pixel) spectra

Given a hyperspectral image $I \in \mathbb{R}^{H \times W \times C}$, train a neural network Φ_θ, such that $\Phi_\theta : (x, y) \mapsto I(x, y)$

Model training



Ground truth pixel spectra $I(x, y)$

Reconstruction loss between predicted pixel $\hat{I}(x, y)$ spectra and ground truth pixel spectra $I(x, y)$

Location (x, y)

MLP with Sinosoidal Activation Functions

Predicted pixel spectra $\hat{I}(x, y)$

HSI Compression using INRs

Compression

Given a hyperspectral image $I \in \mathbb{R}^{H \times W \times C}$, train a neural network $\Phi_\theta: (x, y) \mapsto I(x, y)$

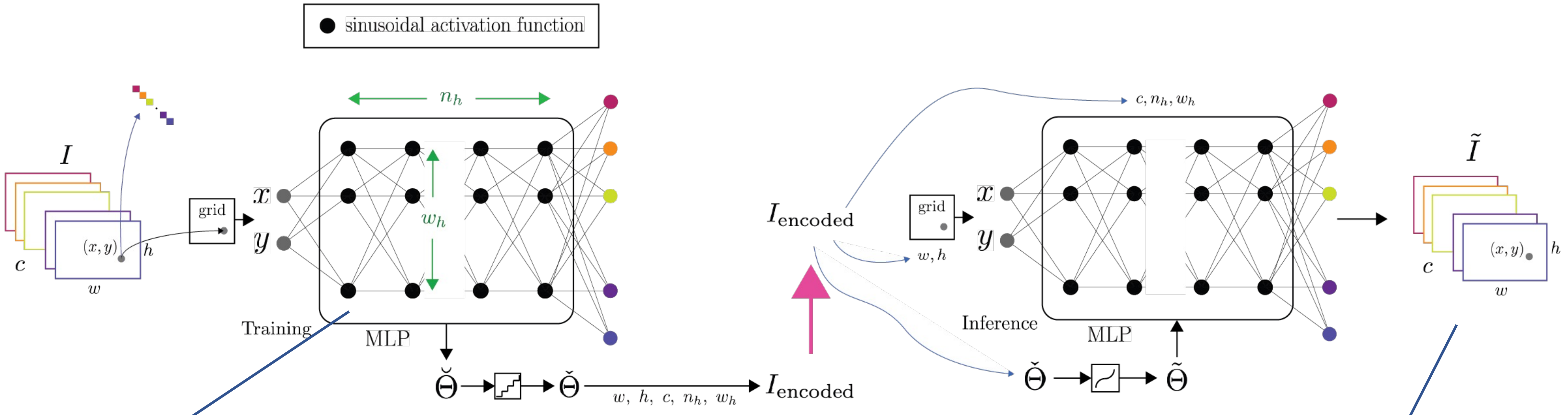
Store network parameters $\theta \in \mathbb{R}^D$ as the representation for I

Decompression

Evaluate $\Phi_\theta(x, y)$ at pixel locations to reconstruct the image

θ is a compressed encoding of image I since $D \ll (H)(W)(C)$

Quantize θ to achieve further savings



A multilayer perceptron network f_Θ with sinusoidal activation functions “learns” to map pixel locations to pixel intensities for a given hyperspectral image I.

The parameters of the network, along with its structure, represent a compressed encoding of the original hyperspectral image.

To reconstruct the hyperspectral image, the transmitted MLP is evaluated at all pixel positions.

HSI Compression using INRs

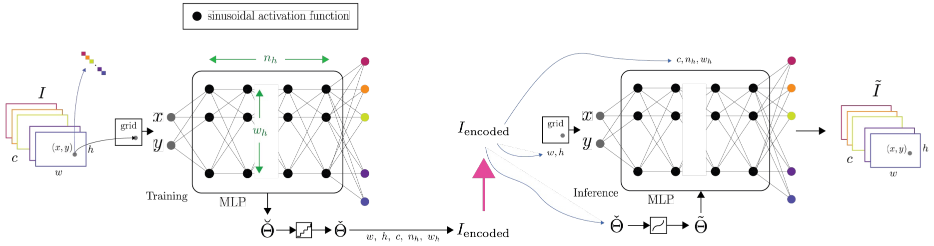
A thought experiment

A 100x100, 300 channel HSI image

100 x 100 x 300 x 4
= 12 MB

INR representation

10 hidden layers with width 30
= (4 + 4 + 4)
+ (4 + 4)
+ (3 x 30) + (9 x 31 x 30) + (300 x 30) x 4
= 0.07 MB



HSI Compression using INRs

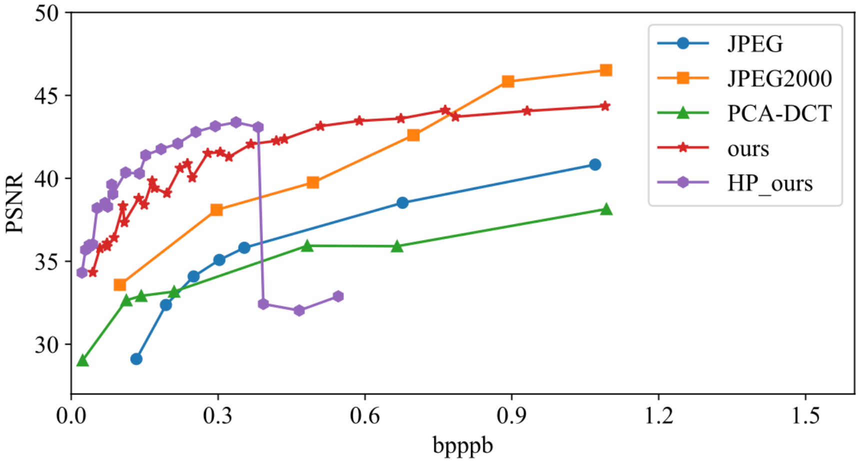
Question 1: Is it possible to achieve high compression rates while maintaining acceptable quality when using implicit neural representations?

PSNR at different compression rates

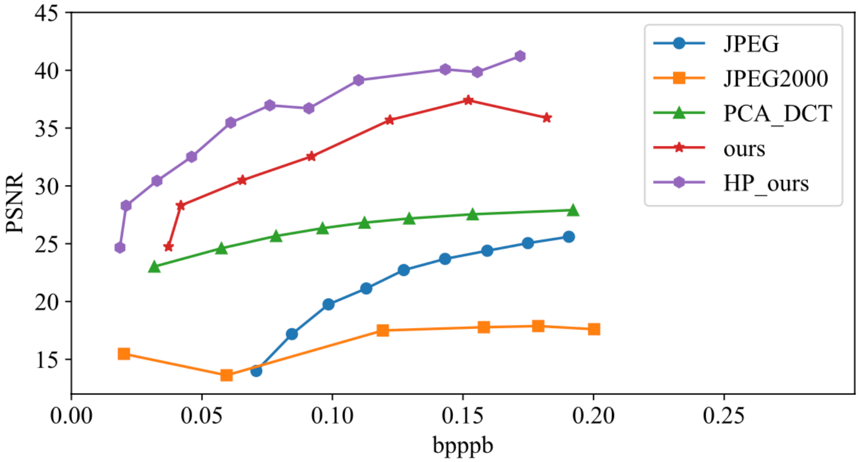
bppppb = 8 for uncompressed images

Smaller bppppb reflects higher compression rates

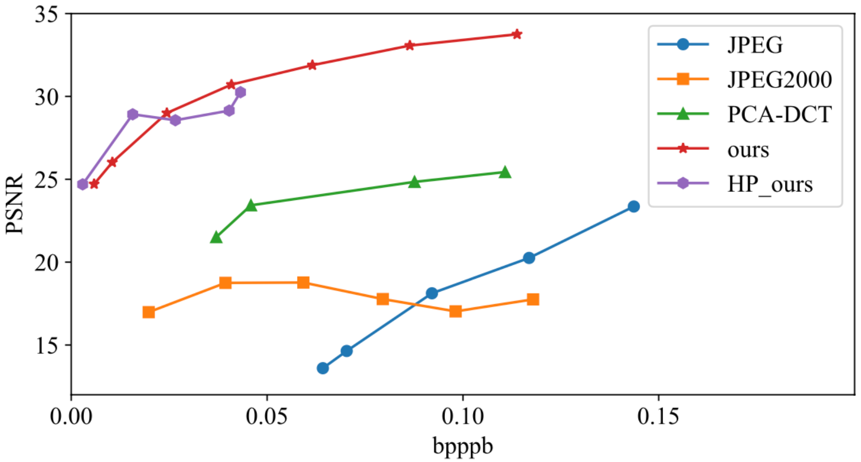
For INR, network structures determines bppppb



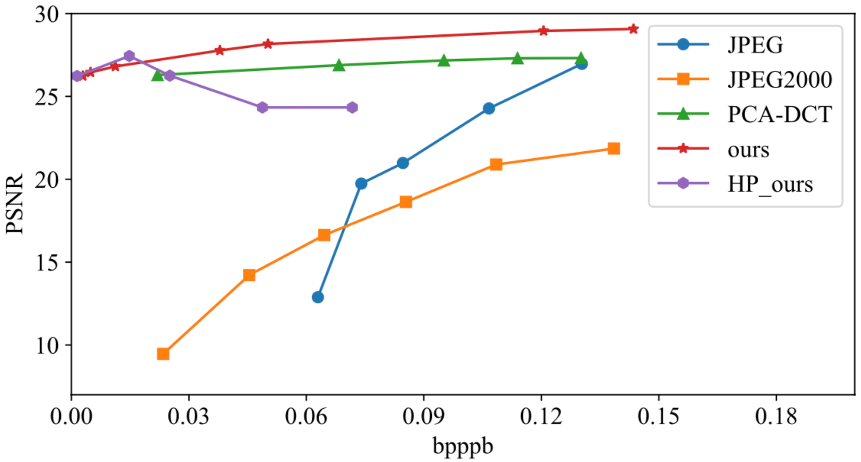
Indian Pines



Jasper Ridge

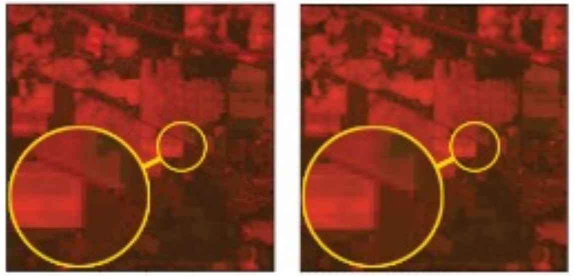


Pavia University



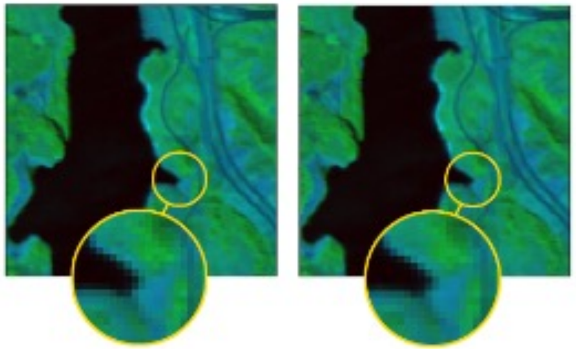
Cuprite

Qualitative results



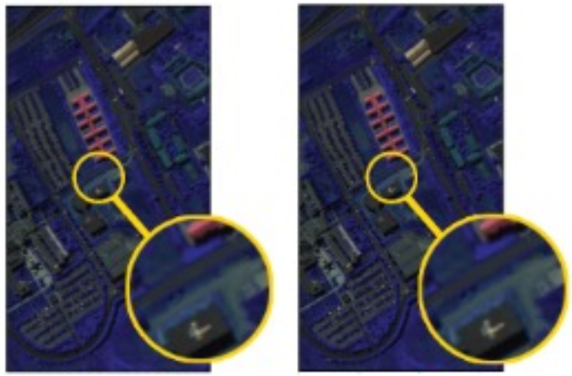
Original Reconstructed

Indian Pines



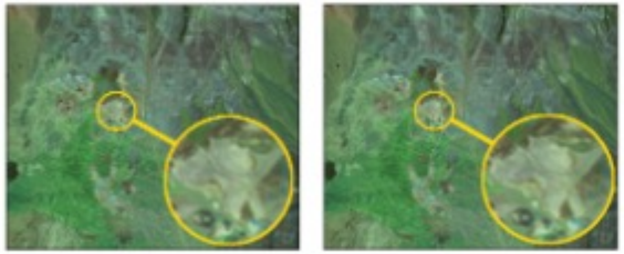
Original Reconstructed

Jasper Ridge



Original Reconstructed

Pavia University



Original Reconstructed

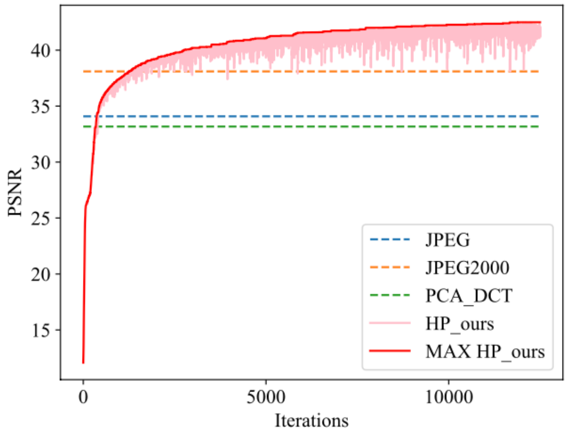
Cuprite

HSI Compression using INRs

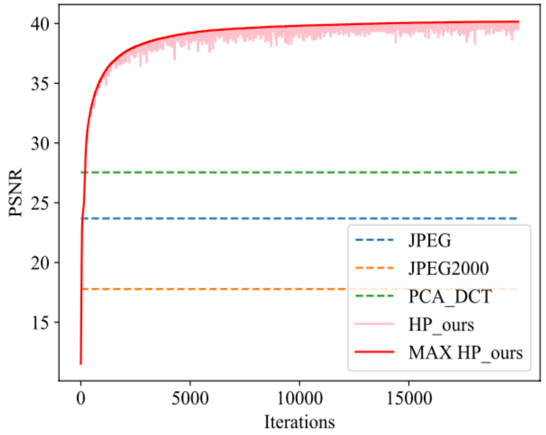
Question 1: Is it possible to achieve high compression rates while maintaining acceptable quality when using implicit neural representations?

YES

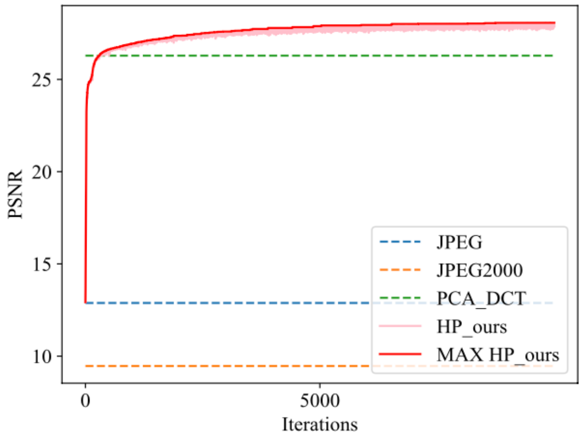
Model training (PSNR vs. Epochs)



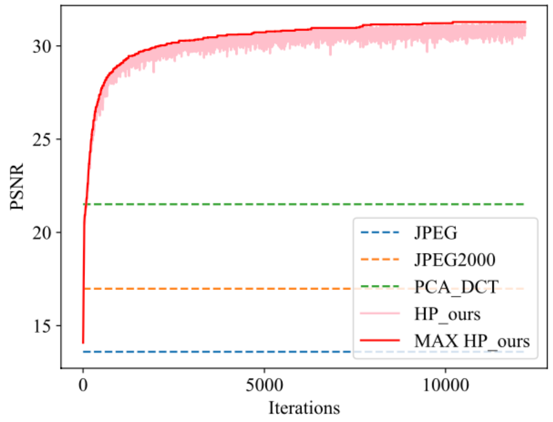
Indian Pines



Jasper Ridge



Cuprite



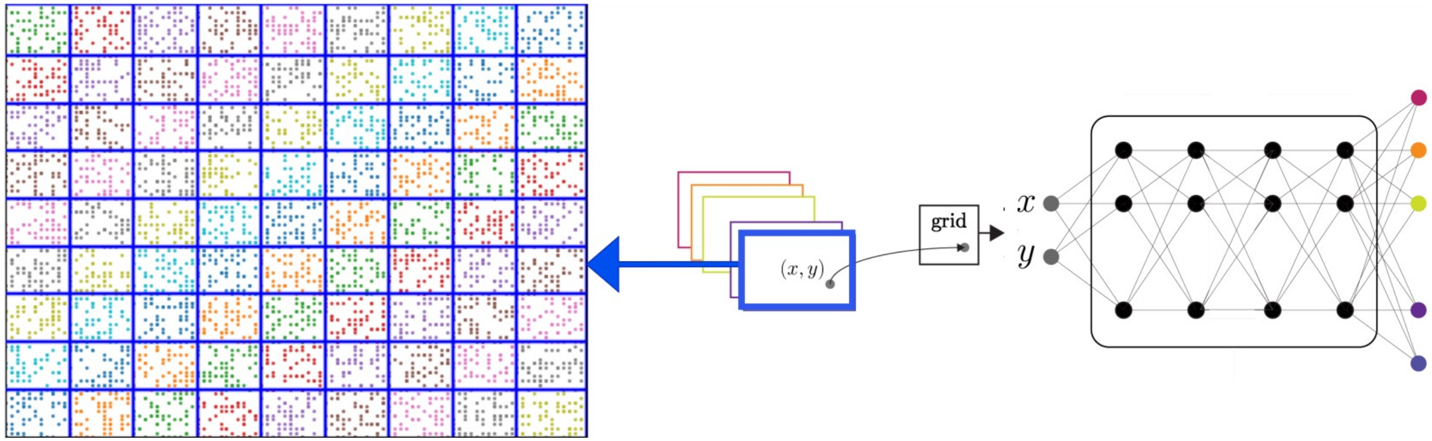
Pavia University

Compression is a slow process since it require multiple training epochs.

Architecture search further slows down the process.

Reducing compression times

Proposal: Do not visit every pixel location during training. Rather employ *sampling*.



An image is divided into tiles and each fraction of pixels are sampled within each tile

Reducing compression times

Proposal: Do not visit every pixel location during training. Rather employ *sampling*.

Question 2: Is it possible to achieve high compression rates while maintaining acceptable quality when using *sampling*?

Comparison (compression times)

Dataset	Method	bppppb	compression time (Sec)	decompression time (Sec)	PSNR ↑
Indian Pines	ours	0.1	243.64	0	36.98
	hp_ours	0.05	243.64	0	36.95
	ours_sampling	0.1	132.87	0.0005	**39.20**
	hp_ours_sampling	0.05	132.87	0.0005	29.94
	JPEG	0.1	7.353	3.27	27.47
	JPEG2000	0.1	0.1455	0.3115	33.58
	PCA-DCT	0.1	1.66	0.04	32.28
Jasper Ridge	ours	0.1	235.19	0.0005	35.77
	hp_ours	0.06	235.19	0.0005	35.70
	ours_sampling	0.1	126.33	0.0005	**40.20**
	hp_ours_sampling	0.06	126.33	0.0005	19.58
	JPEG	0.1	3.71	1.62	24.39
	JPEG2000	0.1	0.138	0.395	16.75
	PCA-DCT	0.1	1.029	0.027	25.98
Pavia University	ours	0.1	352.74	0.0009	33.67
	hp_ours	0.05	352.74	0.0009	19.75
	ours_sampling	0.1	72.512	0.0004	**38.08**
	hp_ours_sampling	0.05	72.512	0.0004	27.02
	JPEG	0.1	33.86	14.61	20.86
	JPEG2000	0.1	0.408	0.628	17.02
	PCA-DCT	0.1	6.525	0.235	25.121
Cuprite	ours	0.06	1565.97	0.001	28.02
	hp_ours	0.03	1565.97	0.001	27.90
	ours_sampling	0.06	664.87	0.001	**37.27**
	hp_ours_sampling	0.03	664.87	0.001	24.85
	JPEG	0.06	101.195	45.02	12.88
	JPEG2000	0.06	1.193	2.476	15.16
	PCA-DCT	0.06	11.67	0.754	26.75

Comparison (PSNR)

Dataset	Method	bppppb	compression time (Sec)	decompression time (Sec)	PSNR ↑
Indian Pines	ours	0.1	243.64	0	36.98
	hp_ours	0.05	243.64	0	36.95
	ours_sampling	0.1	132.87	0.0005	**39.20**
	hp_ours_sampling	0.05	132.87	0.0005	29.94
	JPEG	0.1	7.353	3.27	27.47
	JPEG2000	0.1	0.1455	0.3115	33.58
	PCA-DCT	0.1	1.66	0.04	32.28
Jasper Ridge	ours	0.1	235.19	0.0005	35.77
	hp_ours	0.06	235.19	0.0005	35.70
	ours_sampling	0.1	126.33	0.0005	**40.20**
	hp_ours_sampling	0.06	126.33	0.0005	19.58
	JPEG	0.1	3.71	1.62	24.39
	JPEG2000	0.1	0.138	0.395	16.75
	PCA-DCT	0.1	1.029	0.027	25.98
Pavia University	ours	0.1	352.74	0.0009	33.67
	hp_ours	0.05	352.74	0.0009	19.75
	ours_sampling	0.1	72.512	0.0004	**38.08**
	hp_ours_sampling	0.05	72.512	0.0004	27.02
	JPEG	0.1	33.86	14.61	20.86
	JPEG2000	0.1	0.408	0.628	17.02
	PCA-DCT	0.1	6.525	0.235	25.121
Cuprite	ours	0.06	1565.97	0.001	28.02
	hp_ours	0.03	1565.97	0.001	27.90
	ours_sampling	0.06	664.87	0.001	**37.27**
	hp_ours_sampling	0.03	664.87	0.001	24.85
	JPEG	0.06	101.195	45.02	12.88
	JPEG2000	0.06	1.193	2.476	15.16
	PCA-DCT	0.06	11.67	0.754	26.75

Baseline Methods

Comparison baseline methods used for evaluating the proposed compression approach.

Dataset	Method	References
Indian Pines	PCA+JPEG2000	[Du et al., 2007]
	FPCA+JPEG2000	[Mei et al., 2018]
	HEVC	[Sullivan et al., 2012]
	RPM	[Paul et al., 2016]
Cuprite	PCA+JPEG2000	[Du et al., 2007]
	3D DCT	[Yadav et al., 2018]
	3D DWT+SVR	[Zikiou et al., 2021]
	WSRC	[Ouahioune et al., 2021]
	HEVC	[Sullivan et al., 2012]
	RPM	[Paul et al., 2016]
	3D DCT	[Yadav et al., 2018]
	3D-SPECK	[Tang et al., 2006]
	3D-SPHIT	[Fowler et al., 2007]
	3D- WBTC	[Bajpai et al., 2019]
	3D-LSK	[Ngadiran et al., 2010]
	3D-NLS	[Sudha et al., 2013]
	3D-LMBTC	[Bajpai et al., 2020]
	3D-ZM-SPECK	[Bajpai et al., 2022]

Comparison (quality)

Method	Dataset	Size (KB)	PSNR	bpppb	n_h, w_h	Dataset	Size (KB)	PSNR	bpppb	n_h, w_h
-	Indian Pines	9251	∞	16	-,-	Jasper Ridge	4800	∞	16	-,-
JPEG		115.6	34.085	0.2	-,-		30	21.130	0.1	-,-
JPEG2000		115.6	38.098	0.2	-,-		30	17.494	0.1	-,-
PCA-DCT		115.6	33.173	0.2	-,-		30	26.821	0.1	-,-
PCA+JPEG2000		115.6	39.5	0.2	-,-		30	-	0.1	-,-
FPCA+JPEG2000		115.6	40.5	0.2	-,-		30	-	0.1	-,-
HEVC		115.6	32	0.2	-,-		30	-	0.1	-,-
RPM		115.6	38	0.2	-,-		30	-	0.1	-,-
3D SPECK		115.6	-	0.2	-,-		30	-	0.1	-,-
3D DCT		115.6	-	0.2	-,-		30	-	0.1	-,-
3D DWT+SVR		115.6	-	0.2	-,-		30	-	0.1	-,-
WSRC		115.6	-	0.2	-,-		30	-	0.1	-,-
ours		115.6	40.61	0.2	15,40		30	35.696	0.1	10,20
hp_ours		57.5	40.35	0.1	15,40		15	35.467	0.06	10,20
ours_sampling		115.6	**44.46**	0.2	15,40		30	**41.58**	0.1	15,20
hp_ours_sampling		57.5	30.20	0.2	15,40		15	21.48	0.06	15,20
-	Pavia University	42724	∞	16	-,-	Cuprite	140836	∞	16	-,-
JPEG		267	20.253	0.1	-,-		880.2	24.274	0.1	-,-
JPEG2000		267	17.752	0.1	-,-		880.2	20.889	0.1	-,-
PCA-DCT		267	25.436	0.1	-,-		880.2	27.302	0.1	-,-
PCA+JPEG2000		267	-	0.1	-,-		880.2	27.5	0.1	-,-
FPCA+JPEG2000		267	-	0.1	-,-		880.2	-	0.1	-,-
HEVC		267	-	0.1	-,-		880.2	31	0.1	-,-
RPM		267	-	0.1	-,-		880.2	34	0.1	-,-
3D SPECK		267	-	0.1	-,-		880.2	27.1	0.1	-,-
3D DCT		267	-	0.1	-,-		880.2	33.4	0.1	-,-
3D DWT+SVR		267	-	0.1	-,-		880.2	28.20	0.1	-,-
WSRC		267	-	0.1	-,-		880.2	35	0.1	-,-
ours		267	33.749	0.1	20,60		880.2	28.954	0.1	25,100
hp_ours		133.5	20.886	0.05	20,60		440.1	24.334	0.06	25,100
ours_sampling		267	**40.001**	0.1	10,100		880.2	**37.007**	0.1	25,100
hp_ours_sampling		133.5	27.49	0.05	10,100		440.1	24.96	0.06	25,100

Comparison (quality)

Method	Dataset	Size (KB)	PSNR	bpppb	n_h, w_h	Dataset	Size (KB)	PSNR	bpppb	n_h, w_h
-	Indian Pines	9251	∞	16	-,-	Jasper Ridge	4800	∞	16	-,-
JPEG		115.6	34.085	0.2	-,-		30	21.130	0.1	-,-
JPEG2000		115.6	38.098	0.2	-,-		30	17.494	0.1	-,-
PCA-DCT		115.6	33.173	0.2	-,-		30	26.821	0.1	-,-
PCA+JPEG2000		115.6	39.5	0.2	-,-		30	-	0.1	-,-
FPCA+JPEG2000		115.6	40.5	0.2	-,-		30	-	0.1	-,-
HEVC		115.6	32	0.2	-,-		30	-	0.1	-,-
RPM		115.6	38	0.2	-,-		30	-	0.1	-,-
3D SPECK		115.6	-	0.2	-,-		30	-	0.1	-,-
3D DCT		115.6	-	0.2	-,-		30	-	0.1	-,-
3D DWT+SVR		115.6	-	0.2	-,-		30	-	0.1	-,-
WSRC		115.6	-	0.2	-,-		30	-	0.1	-,-
ours		115.6	40.61	0.2	15,40		30	35.696	0.1	10,20
hp_ours		57.5	40.35	0.1	15,40		15	35.467	0.06	10,20
ours_sampling		115.6	**44.46**	0.2	15,40		30	**41.58**	0.1	15,20
hp_ours_sampling		57.5	30.20	0.2	15,40		15	21.48	0.06	15,20
-	Pavia University	42724	∞	16	-,-	Cuprite	140836	∞	16	-,-
JPEG		267	20.253	0.1	-,-		880.2	24.274	0.1	-,-
JPEG2000		267	17.752	0.1	-,-		880.2	20.889	0.1	-,-
PCA-DCT		267	25.436	0.1	-,-		880.2	27.302	0.1	-,-
PCA+JPEG2000		267	-	0.1	-,-		880.2	27.5	0.1	-,-
FPCA+JPEG2000		267	-	0.1	-,-		880.2	-	0.1	-,-
HEVC		267	-	0.1	-,-		880.2	31	0.1	-,-
RPM		267	-	0.1	-,-		880.2	34	0.1	-,-
3D SPECK		267	-	0.1	-,-		880.2	27.1	0.1	-,-
3D DCT		267	-	0.1	-,-		880.2	33.4	0.1	-,-
3D DWT+SVR		267	-	0.1	-,-		880.2	28.20	0.1	-,-
WSRC		267	-	0.1	-,-		880.2	35	0.1	-,-
ours		267	33.749	0.1	20,60		880.2	28.954	0.1	25,100
hp_ours		133.5	20.886	0.05	20,60		440.1	24.334	0.06	25,100
ours_sampling		267	**40.001**	0.1	10,100		880.2	**37.007**	0.1	25,100
hp_ours_sampling		133.5	27.49	0.05	10,100		440.1	24.96	0.06	25,100

Comparison (quality)

Method	Dataset	Size (KB)	PSNR	bpppb	n_h, w_h	Dataset	Size (KB)	PSNR	bpppb	n_h, w_h
-	~~Indian Pines~~	9251	∞	16	-,-	Jasper Ridge	4800	∞	16	-,-
JPEG		115.6	34.085	0.2	-,-		30	21.130	0.1	-,-
JPEG2000		115.6	38.098	0.2	-,-		30	17.494	0.1	-,-
PCA-DCT		115.6	33.173	0.2	-,-		30	26.821	0.1	-,-
PCA+JPEG2000		115.6	39.5	0.2	-,-		30	-	0.1	-,-
FPCA+JPEG2000		115.6	40.5	0.2	-,-		30	-	0.1	-,-
HEVC		115.6	32	0.2	-,-		30	-	0.1	-,-
RPM		115.6	38	0.2	-,-		30	-	0.1	-,-
3D SPECK		115.6	-	0.2	-,-		30	-	0.1	-,-
3D DCT		115.6	-	0.2	-,-		30	-	0.1	-,-
3D DWT+SVR		115.6	-	0.2	-,-		30	-	0.1	-,-
WSRC		115.6	-	0.2	-,-		30	-	0.1	-,-
ours		115.6	40.61	0.2	15,40		30	35.696	0.1	10,20
hp_ours		57.5	40.35	0.1	15,40		15	35.467	0.06	10,20
ours_sampling		115.6	~~**44.46**~~	0.2	15,40		30	**41.58**	0.1	15,20
hp_ours_sampling		57.5	30.20	0.2	15,40		15	21.48	0.06	15,20
-	Pavia University	42724	∞	16	-,-	Cuprite	140836	∞	16	-,-
JPEG		267	20.253	0.1	-,-		880.2	24.274	0.1	-,-
JPEG2000		267	17.752	0.1	-,-		880.2	20.889	0.1	-,-
PCA-DCT		267	25.436	0.1	-,-		880.2	27.302	0.1	-,-
PCA+JPEG2000		267	-	0.1	-,-		880.2	27.5	0.1	-,-
FPCA+JPEG2000		267	-	0.1	-,-		880.2	-	0.1	-,-
HEVC		267	-	0.1	-,-		880.2	31	0.1	-,-
RPM		267	-	0.1	-,-		880.2	34	0.1	-,-
3D SPECK		267	-	0.1	-,-		880.2	27.1	0.1	-,-
3D DCT		267	-	0.1	-,-		880.2	33.4	0.1	-,-
3D DWT+SVR		267	-	0.1	-,-		880.2	28.20	0.1	-,-
WSRC		267	-	0.1	-,-		880.2	35	0.1	-,-
ours		267	33.749	0.1	20,60		880.2	28.954	0.1	25,100
hp_ours		133.5	20.886	0.05	20,60		440.1	24.334	0.06	25,100
ours_sampling		267	**40.001**	0.1	10,100		880.2	**37.007**	0.1	25,100
hp_ours_sampling		133.5	27.49	0.05	10,100		440.1	24.96	0.06	25,100

Comparison (quality)

Method	Dataset	Size (KB)	PSNR	bpppb	n_h, w_h	Dataset	Size (KB)	PSNR	bpppb	n_h, w_h
-	Indian Pines	9251	∞	16	-,-	Jasper Ridge	4800	∞	16	-,-
JPEG		115.6	34.085	0.2	-,-		30	21.130	0.1	-,-
JPEG2000		115.6	38.098	0.2	-,-		30	17.494	0.1	-,-
PCA-DCT		115.6	33.173	0.2	-,-		30	26.821	0.1	-,-
PCA+JPEG2000		115.6	39.5	0.2	-,-		30	-	0.1	-,-
FPCA+JPEG2000		115.6	40.5	0.2	-,-		30	-	0.1	-,-
HEVC		115.6	32	0.2	-,-		30	-	0.1	-,-
RPM		115.6	38	0.2	-,-		30	-	0.1	-,-
3D SPECK		115.6	-	0.2	-,-		30	-	0.1	-,-
3D DCT		115.6	-	0.2	-,-		30	-	0.1	-,-
3D DWT+SVR		115.6	-	0.2	-,-		30	-	0.1	-,-
WSRC		115.6	-	0.2	-,-		30	-	0.1	-,-
ours		115.6	40.61	0.2	15,40		30	35.696	0.1	10,20
hp_ours		57.5	40.35	0.1	15,40		15	35.467	0.06	10,20
ours_sampling		115.6	**44.46**	0.2	15,40		30	**41.58**	0.1	15,20
hp_ours_sampling		57.5	30.20	0.2	15,40		15	21.48	0.06	15,20
-	~~Pavia University~~	42724	∞	16	-,-	Cuprite	140836	∞	16	-,-
JPEG		267	20.253	0.1	-,-		880.2	24.274	0.1	-,-
JPEG2000		267	17.752	0.1	-,-		880.2	20.889	0.1	-,-
PCA-DCT		267	25.436	0.1	-,-		880.2	27.302	0.1	-,-
PCA+JPEG2000		267	-	0.1	-,-		880.2	27.5	0.1	-,-
FPCA+JPEG2000		267	-	0.1	-,-		880.2	-	0.1	-,-
HEVC		267	-	0.1	-,-		880.2	31	0.1	-,-
RPM		267	-	0.1	-,-		880.2	34	0.1	-,-
3D SPECK		267	-	0.1	-,-		880.2	27.1	0.1	-,-
3D DCT		267	-	0.1	-,-		880.2	33.4	0.1	-,-
3D DWT+SVR		267	-	0.1	-,-		880.2	28.20	0.1	-,-
WSRC		267	-	0.1	-,-		880.2	35	0.1	-,-
ours		267	33.749	0.1	20,60		880.2	28.954	0.1	25,100
hp_ours		133.5	20.886	0.05	20,60		440.1	24.334	0.06	25,100
ours_sampling		267	~~**40.001**~~	0.1	10,100		880.2	**37.007**	0.1	25,100
hp_ours_sampling		133.5	27.49	0.05	10,100		440.1	24.96	0.06	25,100

SSIM Metric*

bpppb	method	SSIM ↑
0.1	WSRC	0.75
	ours_sampling	0.9798
0.01	3D-SPECK	0.142
	3D-SPIHT	0.136
	3D-WBTC	0.141
	3D-LSK	0.138
	3D-NLS	0.135
	3D-LMBTC	0.140
	3D-ZM-SPECK	0.141
	ours	0.9565
	hp_ours	0.9514
	ours_sampling	0.9527
	hp_ours_sampling	0.9390

Cuprite

bpppb	method	SSIM ↑
0.1	3D-SPHIT	0.4
	3D-DCT	0.85
	ours	0.9564
	hp_ours	0.9527
	ours_sampling	0.9901
	hp_ours_sampling	0.8518

Pavia University

* Only these baselines provide SSIM scores on the selected benchmarks

SSIM Metric*

bpppb	method	SSIM ↑
0.1	WSRC	0.75
	ours_sampling	0.9798
0.01	3D-SPECK	0.142
	3D-SPIHT	0.136
	3D-WBTC	0.141
	3D-LSK	0.138
	3D-NLS	0.135
	3D-LMBTC	0.140
	3D-ZM-SPECK	0.141
	ours	0.9565
	hp_ours	0.9514
	ours_sampling	0.9527
	hp_ours_sampling	0.9390

Cuprite

bpppb	method	SSIM ↑
0.1	3D-SPHIT	0.4
	3D-DCT	0.85
	ours	0.9564
	hp_ours	0.9527
	ours_sampling	0.9901
	hp_ours_sampling	0.8518

Pavia University

* Only these baselines provide SSIM scores on the selected benchmarks

SSIM Metric*

bpppb	method	SSIM ↑
0.1	WSRC	0.75
	ours_sampling	0.9798
0.01	3D-SPECK	0.142
	3D-SPIHT	0.136
	3D-WBTC	0.141
	3D-LSK	0.138
	3D-NLS	0.135
	3D-LMBTC	0.140
	3D-ZM-SPECK	0.141
	ours	0.9565
	hp_ours	0.9514
	ours_sampling	0.9527
	hp_ours_sampling	0.9390

Cuprite

bpppb	method	SSIM ↑
0.1	3D-SPHIT	0.4
	3D-DCT	0.85
	ours	0.9564
	hp_ours	0.9527
	ours_sampling	0.9901
	hp_ours_sampling	0.8518

Pavia University

* Only these baselines provide SSIM scores on the selected benchmarks

SSIM Metric*

bpppb	method	SSIM ↑
0.1	WSRC	0.75
	ours_sampling	0.9798
0.01	3D-SPECK	0.142
	3D-SPIHT	0.136
	3D-WBTC	0.141
	3D-LSK	0.138
	3D-NLS	0.135
	3D-LMBTC	0.140
	3D-ZM-SPECK	0.141
	ours	0.9565
	hp_ours	0.9514
	ours_sampling	0.9527
	hp_ours_sampling	0.9390

Cuprite

bpppb	method	SSIM ↑
0.1	3D-SPHIT	0.4
	3D-DCT	0.85
	ours	0.9564
	hp_ours	0.9527
	ours_sampling	0.9901
	hp_ours_sampling	0.8518

Pavia University

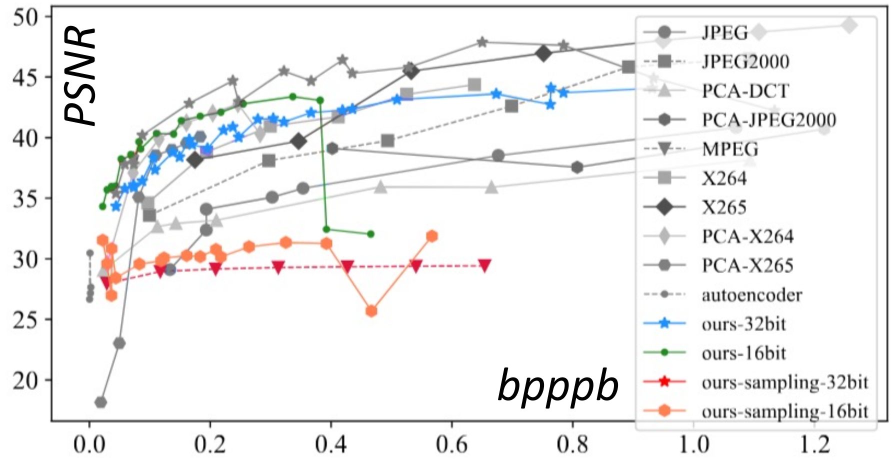
* Only these baselines provide SSIM scores on the selected benchmarks

PSNR at different compression rates

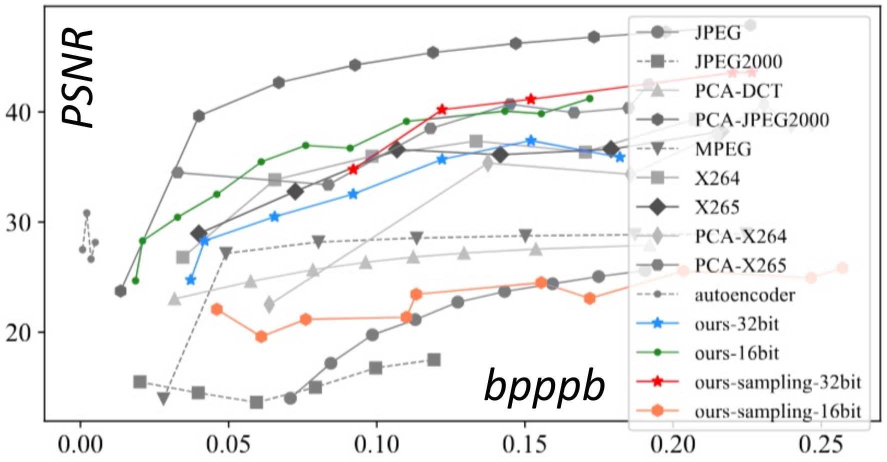
bpppb = 8 for uncompressed images

Smaller bpppb reflects higher compression rates

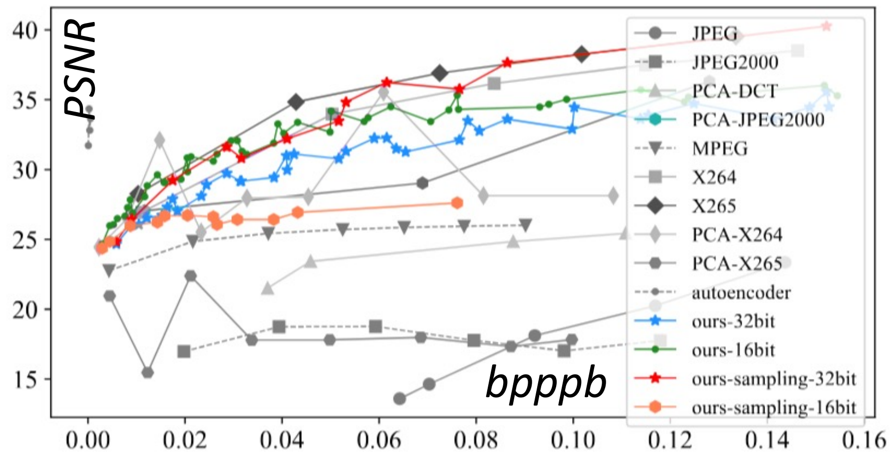
For INR, network structures determines bpppb



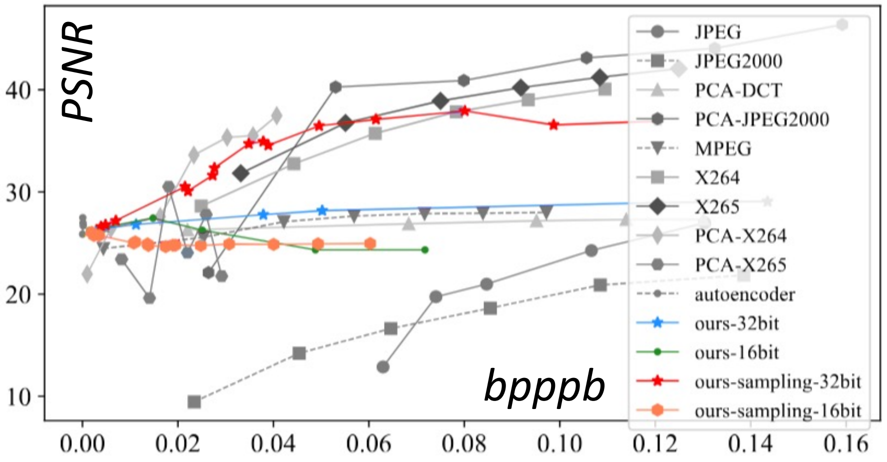
Indian Pines



Jasper Ridge



Pavia University



Cuprite

Reducing compression times

Proposal: Do not visit every pixel location during training. Rather employ *sampling*.

Question 2: Is it possible to achieve high compression rates while maintaining acceptable quality when using *sampling*?

YES

Even faster compression

Network is trained from scratch for each new hyperspectral image

Compression

Given a hyperspectral image $I \in \mathbb{R}^{H\times W\times C}$, train a neural network $\Phi_\theta: (x, y) \mapsto I(x, y)$

Problem

Network training is slow, resulting in large compression times

Why

Does not take advantage of spatial and spectral structural similarities between images

Speeding Up Compression

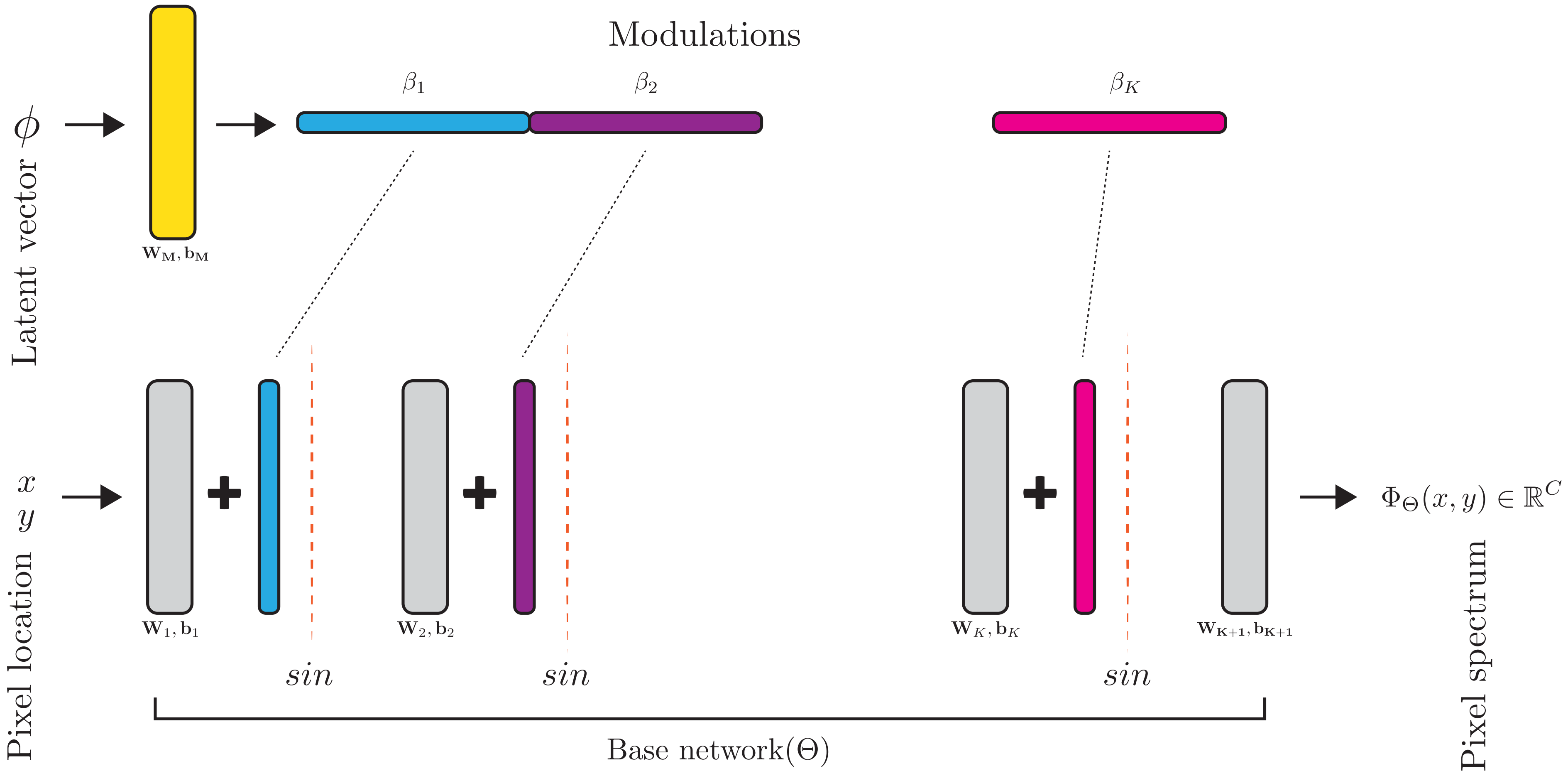
Idea (Meta Learning) [Finn *et al.* 2017, Dupont *et al.* 2022]

A base network encodes the “common” structure of hyperspectral images

Modulations that are applied to the base network record image-specific details

Further savings are achieved by storing a latent code to generate the modulations (i.e., modulations are never explicitly stored)

Meta Network



Speeding Up Compression

Idea (Meta Learning) [Finn *et al.* 2017, Dupont *et al.* 2022]

A base network encodes the "common" structure of hyperspectral images

Modulations that are applied to the base network record image-specific details

Siren Network (an MLP with sinusoidal activations)

Input : $h_0 \in \mathbb{R}^2$

Hidden layers: $h_i = sin(W_i h_{i-1} + b_i)$ $i \in [1, K]$, $W_1 \in \mathbb{R}^{d\times 2}$, $W_i \in \mathbb{R}^{d\times d}$, $b_i \in \mathbb{R}^d$

Output: $h_{K+1} = W_{K+1} h_K + b_{K+1}$ $W_{K+1} \in \mathbb{R}^{C\times d}$, $h_{K+1}, b_{K+1} \in \mathbb{R}^C$

Speeding Up Compression

Idea (Meta Learning)

A base network encodes the "common" structure of hyperspectral images

Modulations that are applied to the base network record image-specific details

Siren Network (an MLP with sinusoidal activations)

Input : $h_0 \in \mathbb{R}^2$

Hidden layers: $h_i = sin(W_i h_{i-1} + b_i)$ $\quad i \in [1, K],\ W_1 \in \mathbb{R}^{d \times 2},\ W_i \in \mathbb{R}^{d \times d},\ b_i \in \mathbb{R}^d$

Output: $h_{K+1} = W_{K+1} h_K + b_{K+1})$ $\quad W_{K+1} \in \mathbb{R}^{C \times d},\ h_{K+1}, b_{K+1} \in \mathbb{R}^C$

Modulations β_i (constructed using latent vector φ)

Modulated hidden layers: $h_i = sin((W_i h_{i-1} + b_i) + \beta_i)$

$$\beta = W_M \varphi + b_M \quad W_M \in \mathbb{R}^{(d)(K) \times d_{latent}},\ \varphi \in \mathbb{R}^{d_{latent}},\ b_M \in \mathbb{R}^{(d)(K)}$$

Speeding Up Compression

Idea (Meta Learning)

A base network encodes the "common" structure of hyperspectral images

Modulations that are applied to the base network record image-specific details

Use

Given a pre-trained network, a new image is "compressed" by updating modulations only

Faster and Cheaper

Plus, we can achieve higher compression by storing only modulations for each image

Cost of the shared network parameters storage is amortized over multiple images

Speeding Up Compression

Idea (Meta Learning)

A base network encodes the “common” structu

Modulations that are applied to the base netw
details

Use

Given a pre-trained network, a new image is “c

Plus, we can achieve higher compression by st

**Cost of the shared network parame
is amortized over multiple images**

A thought experiment

A 100x100, 300 channel HSI image

100 x 100 x 300 x 4
= 12 MB

Only storing modulations

10 hidden layers with width 30
Latent vector size 32
= 32 x 4
= 0.000128 MB

Reducing compression times

Proposal: Exploit spatial and spectral similarities between hyperspectral images using *meta learning* to achieve faster compression

Question 3: Is it possible to achieve faster compression at acceptable PSNR using *meta learning*?

Model Agnostic Meta-Learning

Inner loop

Update image-specific modulations

$$\beta^{(t)} = \beta - \alpha_{inner} \nabla_\beta \mathcal{L}\left(I^{(t)}, \phi_{[\theta|\beta]}\right)$$

Network parameters θ are frozen

Here $\beta^{(t)}$ denotes modulations parameters for image $I^{(t)}$

Initially β are set to 0

Outer loop

Update network parameters θ

$$\theta = \theta - \alpha_{outer} \sum_{t \in [1,T]} \nabla_\theta \mathcal{L}\left(I^{(t)}, \phi_{[\theta|\beta^{(t)}]}\right)$$

$\beta^{(t)}$ is frozen

Latent Vector φ to Construct Modulations β

Inner loop

Update image-specific modulations

$$\varphi^{(t)} = \varphi - \alpha_{inner} \nabla_{\varphi} \mathcal{L}\left(I^{(t)}, \phi_{[\theta^{+}|\varphi]}\right)$$

Set $\theta^{+} = \{\theta, W_M, b_M\}$

Here $\varphi^{(t)}$ denotes latent vector for constructing modulations for image $I^{(t)}$

Initially φ are set to 0

Outer loop

Update network parameters θ and the linear layer for mapping latent vectors to modulations

$$\theta^{+} = \theta^{+} - \alpha_{outer} \sum_{t \in [1,T]} \nabla_{\theta^{+}} \mathcal{L}\left(I^{(t)}, \phi_{[\theta^{+}|\varphi^{(t)}]}\right)$$

Evaluation

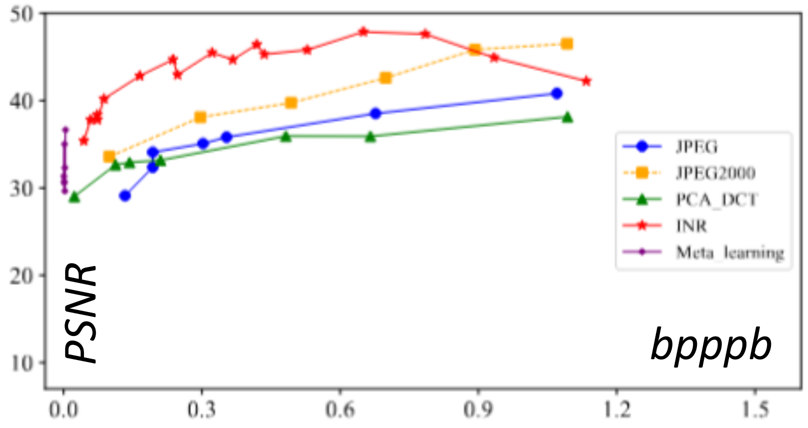
- A single network is trained on four benchmarks
  - Indian Pines
  - Pavia University
  - Jasper Ridge
  - Cuprite
- Modulations capture the structure unique to each benchmark
- Compression time is amortized over four benchmarks

PSNR at different compression rates

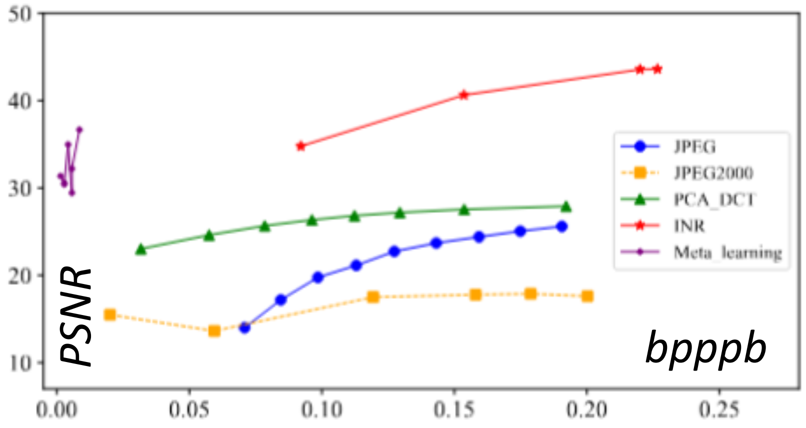
bpppb = 8 for uncompressed images

Smaller bpppb reflects higher compression rates

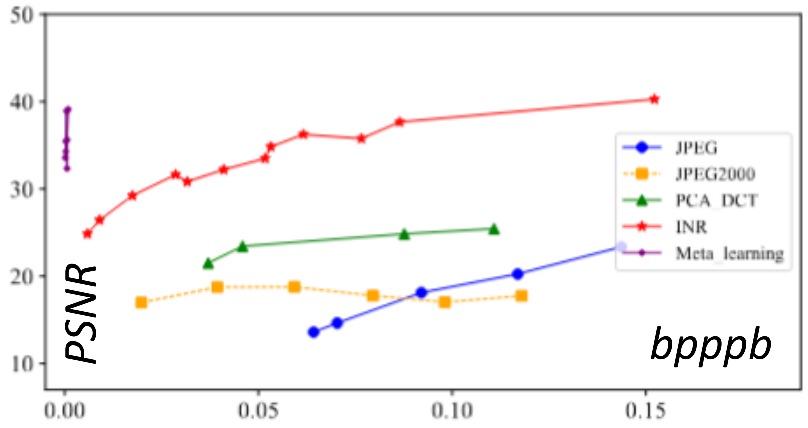
For INR, network structures determines bpppb



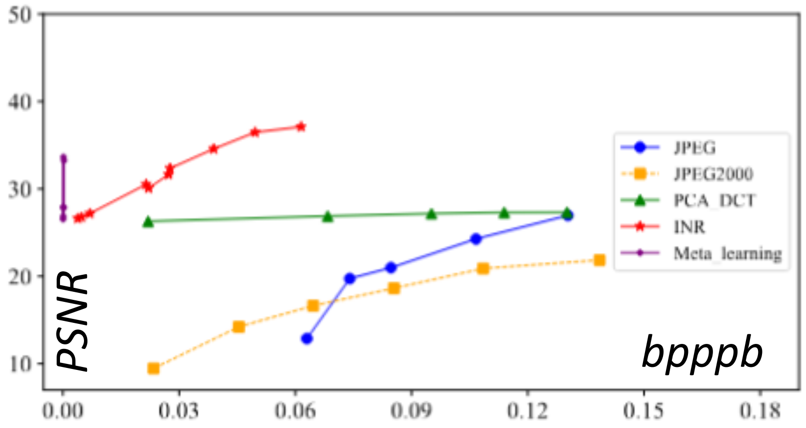
Indian Pines



Jasper Ridge



Pavia University



Cuprite

Comparison (size)

Indian Pines				
Method	Size (KB)	PSNR	bpppb	n_h, w_h
-	9251	∞	16	-,-
JPEG [17, 37]	115.6	34.085	0.2	-,-
JPEG2000 [9]	115.6	35.84	0.2	-,-
PCA-DCT [31]	115.6	33.173	0.2	-,-
PCA+JPEG2000 [9]	115.6	39.5	0.2	-,-
FPCA+JPEG2000 [28]	115.6	40.5	0.2	-,-
HEVC [45]	115.6	32	0.2	-,-
RPM [35]	115.6	38	0.2	-,-
3D SPECK [47]	115.6	-	0.2	-,-
3D DCT [48]	115.6	-	0.2	-,-
3D DWT+SVR [51]	115.6	-	0.2	-,-
WSRC [32]	115.6	-	0.2	-,-
ours-32bit [39]	115.6	42.22	0.2	5,60
ours-16bit [39]	57.5	29.68	0.1	5,60
ours-sampling-32bit [40]	115.6	42.22	0.2	5,60
ours-sampling-16bit [40]	57.5	29.68	0.2	5,60
meta-learning	**0.003**	33.36	6.9e-6	10,128

Jasper Ridge				
Method	Size (KB)	PSNR	bpppb	n_h, w_h
-	4800	∞	16	-,-
JPEG [17, 37]	30	21.130	0.1	-,-
JPEG2000 [9]	30	17.494	0.1	-,-
PCA-DCT [31]	30	26.821	0.1	-,-
PCA+JPEG2000 [9]	30	-	0.1	-,-
FPCA+JPEG2000 [28]	30	-	0.1	-,-
HEVC [45]	30	-	0.1	-,-
RPM [35]	30	-	0.1	-,-
3D SPECK [47]	30	-	0.1	-,-
3D DCT [48]	30	-	0.1	-,-
3D DWT+SVR [51]	30	-	0.1	-,-
WSRC [32]	30	-	0.1	-,-
ours-32bit [39]	30	32.54	0.1	5,20
ours-16bit [39]	15	22.07	0.06	5,20
ours-sampling-32bit [40]	30	34.77	0.1	5,20
ours-sampling-16bit [40]	15	22.07	0.06	5,20
meta-learning	**0.003**	30.87	1.4e-5	10,128

Pavia University				
Method	Size (KB)	PSNR	bpppb	n_h, w_h
-	42724	∞	16	-,-
JPEG [17, 37]	267	20.253	0.1	-,-
JPEG2000 [9]	267	17.752	0.1	-,-
PCA-DCT [31]	267	25.436	0.1	-,-
PCA+JPEG2000 [9]	267	-	0.1	-,-
FPCA+JPEG2000 [28]	267	-	0.1	-,-
HEVC [45]	267	-	0.1	-,-
RPM [35]	267	-	0.1	-,-
3D SPECK [47]	267	-	0.1	-,-
3D DCT [48]	267	-	0.1	-,-
3D DWT+SVR [51]	267	-	0.1	-,-
WSRC [32]	267	-	0.1	-,-
ours-32bit [39]	267	34.46	0.1	10,80
ours-16bit [39]	133.5	34.17	0.05	10,80
ours-sampling-32bit [40]	267	38.08	0.1	10,80
ours-sampling-16bit [40]	133.5	27.49	0.05	10,80
meta-learning	**7.8**	35.53	0.003	10,128

Cuprite				
Method	Size (KB)	PSNR	bpppb	n_h, w_h
-	140836	∞	16	-,-
JPEG [17, 37]	880.2	24.274	0.1	-,-
JPEG2000 [9]	880.2	20.889	0.1	-,-
PCA-DCT [31]	880.2	27.302	0.1	-,-
PCA+JPEG2000 [9]	880.2	40.90	0.1	-,-
FPCA+JPEG2000 [28]	880.2	-	0.1	-,-
HEVC [45]	880.2	31	0.1	-,-
RPM [35]	880.2	34	0.1	-,-
3D SPECK [47]	880.2	27.1	0.1	-,-
3D DCT [48]	880.2	33.4	0.1	-,-
3D DWT+SVR [51]	880.2	28.20	0.1	-,-
WSRC [32]	880.2	35	0.1	-,-
ours-32bit [39]	880.2	28.954	0.1	25,100
ours-16bit [39]	440.1	24.334	0.06	25,100
ours-sampling-32bit [40]	880.2	36.55	0.1	25,90
ours-sampling-16bit [40]	440.1	24.91	0.06	25,90
meta-learning	**0.003**	24.57	4.5e-7	10,128

Indian Pines

Comparison (size)

Indian Pines					Jasper Ridge				
Method	Size (KB)	PSNR	bpppb	n_h, w_h	Method	Size (KB)	PSNR	bpppb	n_h, w_h
-	9251	∞	16	-,-	-	4800	∞	16	-,-
JPEG [17, 37]	115.6	34.085	0.2	-,-	JPEG [17, 37]	30	21.130	0.1	-,-
JPEG2000 [9]	115.6	35.84	0.2	-,-	JPEG2000 [9]	30	17.494	0.1	-,-
PCA-DCT [31]	115.6	33.173	0.2	-,-	PCA-DCT [31]	30	26.821	0.1	-,-
PCA+JPEG2000 [9]	115.6	39.5	0.2	-,-	PCA+JPEG2000 [9]	30	-	0.1	-,-
FPCA+JPEG2000 [28]	115.6	40.5	0.2	-,-	FPCA+JPEG2000 [28]	30	-	0.1	-,-
HEVC [45]	115.6	32	0.2	-,-	HEVC [45]	30	-	0.1	-,-
RPM [35]	115.6	38	0.2	-,-	RPM [35]	30	-	0.1	-,-
3D SPECK [47]	115.6	-	0.2	-,-	3D SPECK [47]	30	-	0.1	-,-
3D DCT [48]	115.6	-	0.2	-,-	3D DCT [48]	30	-	0.1	-,-
3D DWT+SVR [51]	115.6	-	0.2	-,-	3D DWT+SVR [51]	30	-	0.1	-,-
WSRC [32]	115.6	-	0.2	-,-	WSRC [32]	30	-	0.1	-,-
ours-32bit [39]	115.6	42.22	0.2	5,60	ours-32bit [39]	30	32.54	0.1	5,20
ours-16bit [39]	57.5	29.68	0.1	5,60	ours-16bit [39]	15	22.07	0.06	5,20
ours-sampling-32bit [40]	115.6	42.22	0.2	5,60	ours-sampling-32bit [40]	30	34.77	0.1	5,20
ours-sampling-16bit [40]	57.5	29.68	0.2	5,60	ours-sampling-16bit [40]	15	22.07	0.06	5,20
meta-learning	**0.003**	33.36	6.9e-6	10,128	meta-learning	**0.003**	30.87	1.4e-5	10,128
Pavia University					**Cuprite**				
Method	Size (KB)	PSNR	bpppb	n_h, w_h	Method	Size (KB)	PSNR	bpppb	n_h, w_h
-	42724	∞	16	-,-	-	140836	∞	16	-,-
JPEG [17, 37]	267	20.253	0.1	-,-	JPEG [17, 37]	880.2	24.274	0.1	-,-
JPEG2000 [9]	267	17.752	0.1	-,-	JPEG2000 [9]	880.2	20.889	0.1	-,-
PCA-DCT [31]	267	25.436	0.1	-,-	PCA-DCT [31]	880.2	27.302	0.1	-,-
PCA+JPEG2000 [9]	267	-	0.1	-,-	PCA+JPEG2000 [9]	880.2	40.90	0.1	-,-
FPCA+JPEG2000 [28]	267	-	0.1	-,-	FPCA+JPEG2000 [28]	880.2	-	0.1	-,-
HEVC [45]	267	-	0.1	-,-	HEVC [45]	880.2	31	0.1	-,-
RPM [35]	267	-	0.1	-,-	RPM [35]	880.2	34	0.1	-,-
3D SPECK [47]	267	-	0.1	-,-	3D SPECK [47]	880.2	27.1	0.1	-,-
3D DCT [48]	267	-	0.1	-,-	3D DCT [48]	880.2	33.4	0.1	-,-
3D DWT+SVR [51]	267	-	0.1	-,-	3D DWT+SVR [51]	880.2	28.20	0.1	-,-
WSRC [32]	267	-	0.1	-,-	WSRC [32]	880.2	35	0.1	-,-
ours-32bit [39]	267	34.46	0.1	10,80	ours-32bit [39]	880.2	28.954	0.1	25,100
ours-16bit [39]	133.5	34.17	0.05	10,80	ours-16bit [39]	440.1	24.334	0.06	25,100
ours-sampling-32bit [40]	267	38.08	0.1	10,80	ours-sampling-32bit [40]	880.2	36.55	0.1	25,90
ours-sampling-16bit [40]	133.5	27.49	0.05	10,80	ours-sampling-16bit [40]	440.1	24.91	0.06	25,90
meta-learning	**7.8**	35.53	0.003	10,128	meta-learning	**0.003**	24.57	4.5e-7	10,128

Pavia University

Comparison (size)

Indian Pines					Jasper Ridge				
Method	Size (KB)	PSNR	bpppb	n_h, w_h	Method	Size (KB)	PSNR	bpppb	n_h, w_h
-	9251	∞	16	-,-	-	4800	∞	16	-,-
JPEG [17, 37]	115.6	34.085	0.2	-,-	JPEG [17, 37]	30	21.130	0.1	-,-
JPEG2000 [9]	115.6	35.84	0.2	-,-	JPEG2000 [9]	30	17.494	0.1	-,-
PCA-DCT [31]	115.6	33.173	0.2	-,-	PCA-DCT [31]	30	26.821	0.1	-,-
PCA+JPEG2000 [9]	115.6	39.5	0.2	-,-	PCA+JPEG2000 [9]	30	-	0.1	-,-
FPCA+JPEG2000 [28]	115.6	40.5	0.2	-.-	FPCA+JPEG2000 [28]	30	-	0.1	-,-
HEVC [45]	115.6	32	0.2	-,-	HEVC [45]	30	-	0.1	-,-
RPM [35]	115.6	38	0.2	-,-	RPM [35]	30	-	0.1	-,-
3D SPECK [47]	115.6	-	0.2	-,-	3D SPECK [47]	30	-	0.1	-,-
3D DCT [48]	115.6	-	0.2	-,-	3D DCT [48]	30	-	0.1	-,-
3D DWT+SVR [51]	115.6	-	0.2	-,-	3D DWT+SVR [51]	30	-	0.1	-,-
WSRC [32]	115.6	-	0.2	-,-	WSRC [32]	30	-	0.1	-,-
ours-32bit [39]	115.6	42.22	0.2	5,60	ours-32bit [39]	30	32.54	0.1	5,20
ours-16bit [39]	57.5	29.68	0.1	5,60	ours-16bit [39]	15	22.07	0.06	5,20
ours-sampling-32bit [40]	115.6	42.22	0.2	5,60	ours-sampling-32bit [40]	30	34.77	0.1	5,20
ours-sampling-16bit [40]	57.5	29.68	0.2	5,60	ours-sampling-16bit [40]	15	22.07	0.06	5,20
meta-learning	**0.003**	33.36	6.9e-6	10,128	meta-learning	**0.003**	30.87	1.4e-5	10,128

Pavia University					Cuprite				
Method	Size (KB)	PSNR	bpppb	n_h, w_h	Method	Size (KB)	PSNR	bpppb	n_h, w_h
-	42724	∞	16	-,-	-	140836	∞	16	-,-
JPEG [17, 37]	267	20.253	0.1	-,-	JPEG [17, 37]	880.2	24.274	0.1	-,-
JPEG2000 [9]	267	17.752	0.1	-,-	JPEG2000 [9]	880.2	20.889	0.1	-,-
PCA-DCT [31]	267	25.436	0.1	-,-	PCA-DCT [31]	880.2	27.302	0.1	-,-
PCA+JPEG2000 [9]	267	-	0.1	-,-	PCA+JPEG2000 [9]	880.2	40.90	0.1	-,-
FPCA+JPEG2000 [28]	267	-	0.1	-,-	FPCA+JPEG2000 [28]	880.2	-	0.1	-,-
HEVC [45]	267	-	0.1	-,-	HEVC [45]	880.2	31	0.1	-,-
RPM [35]	267	-	0.1	-,-	RPM [35]	880.2	34	0.1	-,-
3D SPECK [47]	267	-	0.1	-,-	3D SPECK [47]	880.2	27.1	0.1	-,-
3D DCT [48]	267	-	0.1	-,-	3D DCT [48]	880.2	33.4	0.1	-,-
3D DWT+SVR [51]	267	-	0.1	-,-	3D DWT+SVR [51]	880.2	28.20	0.1	-,-
WSRC [32]	267	-	0.1	-,-	WSRC [32]	880.2	35	0.1	-,-
ours-32bit [39]	267	34.46	0.1	10,80	ours-32bit [39]	880.2	28.954	0.1	25,100
ours-16bit [39]	133.5	34.17	0.05	10,80	ours-16bit [39]	440.1	24.334	0.06	25,100
ours-sampling-32bit [40]	267	38.08	0.1	10,80	ours-sampling-32bit [40]	880.2	36.55	0.1	25,90
ours-sampling-16bit [40]	133.5	27.49	0.05	10,80	ours-sampling-16bit [40]	440.1	24.91	0.06	25,90
meta-learning	**7.8**	35.53	0.003	10,128	meta-learning	**0.003**	24.57	4.5e-7	10,128

Jasper Ridge

Comparison (size)

Indian Pines					Jasper Ridge				
Method	Size (KB)	PSNR	bpppb	n_h, w_h	Method	Size (KB)	PSNR	bpppb	n_h, w_h
-	9251	∞	16	-,-	-	4800	∞	16	-,-
JPEG [17, 37]	115.6	34.085	0.2	-,-	JPEG [17, 37]	30	21.130	0.1	-,-
JPEG2000 [9]	115.6	35.84	0.2	-,-	JPEG2000 [9]	30	17.494	0.1	-,-
PCA-DCT [31]	115.6	33.173	0.2	-,-	PCA-DCT [31]	30	26.821	0.1	-,-
PCA+JPEG2000 [9]	115.6	39.5	0.2	-,-	PCA+JPEG2000 [9]	30	-	0.1	-,-
FPCA+JPEG2000 [28]	115.6	40.5	0.2	-.-	FPCA+JPEG2000 [28]	30	-	0.1	-,-
HEVC [45]	115.6	32	0.2	-,-	HEVC [45]	30	-	0.1	-,-
RPM [35]	115.6	38	0.2	-,-	RPM [35]	30	-	0.1	-,-
3D SPECK [47]	115.6	-	0.2	-,-	3D SPECK [47]	30	-	0.1	-,-
3D DCT [48]	115.6	-	0.2	-,-	3D DCT [48]	30	-	0.1	-,-
3D DWT+SVR [51]	115.6	-	0.2	-,-	3D DWT+SVR [51]	30	-	0.1	-,-
WSRC [32]	115.6	-	0.2	-,-	WSRC [32]	30	-	0.1	-,-
ours-32bit [39]	115.6	42.22	0.2	5,60	ours-32bit [39]	30	32.54	0.1	5,20
ours-16bit [39]	57.5	29.68	0.1	5,60	ours-16bit [39]	15	22.07	0.06	5,20
ours-sampling-32bit [40]	115.6	42.22	0.2	5,60	ours-sampling-32bit [40]	30	34.77	0.1	5,20
ours-sampling-16bit [40]	57.5	29.68	0.2	5,60	ours-sampling-16bit [40]	15	22.07	0.06	5,20
meta-learning	**0.003**	33.36	6.9e-6	10,128	meta-learning	**0.003**	30.87	1.4e-5	10,128

Pavia University					Cuprite				
Method	Size (KB)	PSNR	bpppb	n_h, w_h	Method	Size (KB)	PSNR	bpppb	n_h, w_h
-	42724	∞	16	-,-	-	140836	∞	16	-,-
JPEG [17, 37]	267	20.253	0.1	-,-	JPEG [17, 37]	880.2	24.274	0.1	-,-
JPEG2000 [9]	267	17.752	0.1	-,-	JPEG2000 [9]	880.2	20.889	0.1	-,-
PCA-DCT [31]	267	25.436	0.1	-,-	PCA-DCT [31]	880.2	27.302	0.1	-,-
PCA+JPEG2000 [9]	267	-	0.1	-,-	PCA+JPEG2000 [9]	880.2	40.90	0.1	-,-
FPCA+JPEG2000 [28]	267	-	0.1	-,-	FPCA+JPEG2000 [28]	880.2	-	0.1	-,-
HEVC [45]	267	-	0.1	-,-	HEVC [45]	880.2	31	0.1	-,-
RPM [35]	267	-	0.1	-,-	RPM [35]	880.2	34	0.1	-,-
3D SPECK [47]	267	-	0.1	-,-	3D SPECK [47]	880.2	27.1	0.1	-,-
3D DCT [48]	267	-	0.1	-,-	3D DCT [48]	880.2	33.4	0.1	-,-
3D DWT+SVR [51]	267	-	0.1	-,-	3D DWT+SVR [51]	880.2	28.20	0.1	-,-
WSRC [32]	267	-	0.1	-,-	WSRC [32]	880.2	35	0.1	-,-
ours-32bit [39]	267	34.46	0.1	10,80	ours-32bit [39]	880.2	28.954	0.1	25,100
ours-16bit [39]	133.5	34.17	0.05	10,80	ours-16bit [39]	440.1	24.334	0.06	25,100
ours-sampling-32bit [40]	267	38.08	0.1	10,80	ours-sampling-32bit [40]	880.2	36.55	0.1	25,90
ours-sampling-16bit [40]	133.5	27.49	0.05	10,80	ours-sampling-16bit [40]	440.1	24.91	0.06	25,90
meta-learning	**7.8**	35.53	0.003	10,128	meta-learning	**0.003**	24.57	4.5e-7	10,128

Cuprite

Comparison (quality)

Indian Pines				
Method	Size (KB	PSNR	bpppb	n_h, w_h
-	9251	∞	16	-,-
JPEG (Good et al., 1994; Qiao et al., 2014)	115.6	34.085	0.2	-,-
JPEG2000 (Du and Fowler, 2007)	115.6	36.098	0.2	-,-
PCA-DCT (Nian et al., 2016)	115.6	33.173	0.2	-,-
PCA+JPEG2000 (Du and Fowler, 2007)	115.6	39.5	0.2	-,-
FPCA+JPEG2000 (Mei et al., 2018)	115.6	40.5	0.2	-,-
HEVC (Sullivan et al., 2012)	115.6	32	0.2	-,-
RPM (Paul et al., 2016)	115.6	38	0.2	-,-
3D SPECK (Tang and Pearlman, 2006)	115.6	-	0.2	-,-
3D DCT (Yadav and Nagmode, 2018)	115.6	-	0.2	-,-
3D DWT+SVR (Zikiou et al., 2020)	115.6	-	0.2	-,-
WSRC (Ouahioune et al., 2021)	115.6	-	0.2	[illegible]
INR (Rezasoltani and Qureshi, 2023)	115.6	40.61	0.2	[illegible]
HP_INR (Rezasoltani and Qureshi, 2023)	57.5	40.35	0.1	[illegible]
INR_sampling (Rezasoltani and Qureshi, 2024)	115.6	44.46	0.2	[illegible]
HP_INR_sampling (Rezasoltani and Qureshi, 2024)	57.5	30.20	0.2	[illegible]
Meta_learning	**2.4**	**36.6**	[illegible]	[illegible]

6th best

Jasper Ridge				
Method	Size (KB	PSNR	bpppb	n_h, w_h
-	4800	∞	16	-,-
JPEG (Good et al., 1994; Qiao et al., 2014)	30	21.130	0.1	-,-
JPEG2000 (Du and Fowler, 2007)	30	17.494	0.1	-,-
PCA-DCT (Nian et al., 2016)	30	26.821	0.1	-,-
PCA+JPEG2000 (Du and Fowler, 2007)	30	-	0.1	-,-
FPCA+JPEG2000 (Mei et al., 2018)	30	-	0.1	-,-
HEVC (Sullivan et al., 2012)	30	-	0.1	-,-
RPM (Paul et al., 2016)	30	-	0.1	-,-
3D SPECK (Tang and Pearlman, 2006)	30	-	0.1	-,-
3D DCT (Yadav and Nagmode, 2018)	30	-	0.1	-,-
3D DWT+SVR (Zikiou et al., 2020)	30	-	0.1	-,-
WSRC (Ouahioune et al., 2021)	30	-	0.1	[illegible]
...R (Rezasoltani and Qureshi, 2023)	30	35.696	0.1	[illegible]
...INR (Rezasoltani and Qureshi, 2023)	15	35.467	0.06	[illegible]
...mpling (Rezasoltani and Qureshi, 2024)	30	41.58	0.1	[illegible]
...sampling (Rezasoltani and Qureshi, 2024)	15	21.48	0.06	[illegible]
Meta_learning	**2.3**	**36.6**	[illegible]	[illegible]

Best

Pavia University				
Method	Size (KB	PSNR	bpppb	n_h, w_h
-	42724	∞	16	-,-
JPEG (Good et al., 1994; Qiao et al., 2014)	267	20.253	0.1	-,-
JPEG2000 (Du and Fowler, 2007)	267	17.752	0.1	-,-
PCA-DCT (Nian et al., 2016)	267	25.436	0.1	-,-
PCA+JPEG2000 (Du and Fowler, 2007)	267	-	0.1	-,-
FPCA+JPEG2000 (Mei et al., 2018)	267	-	0.1	-,-
HEVC (Sullivan et al., 2012)	267	-	0.1	-,-
RPM (Paul et al., 2016)	267	-	0.1	-,-
3D SPECK (Tang and Pearlman, 2006)	267	-	0.1	-,-
3D DCT (Yadav and Nagmode, 2018)	267	-	0.1	-,-
3D DWT+SVR (Zikiou et al., 2020)	267	-	0.1	-,-
WSRC (Ouahioune et al., 2021)	267	-	0.1	[illegible]
INR (Rezasoltani and Qureshi, 2023)	267	33.749	0.1	[illegible]
HP_INR (Rezasoltani and Qureshi, 2023)	133.5	20.886	0.05	[illegible]
INR_sampling (Rezasoltani and Qureshi, 2024)	267	40.001	0.1	[illegible]
HP_INR_sampling (Rezasoltani and Qureshi, 2024)	133.5	27.49	0.05	[illegible]
Meta_learning	**2.1**	**39.1**	[illegible]	[illegible]

2nd best

Cuprite				
Method	Size (KB	PSNR	bpppb	n_h, w_h
-	140836	∞	16	-,-
JPEG (Good et al., 1994; Qiao et al., 2014)	880.2	24.274	0.1	-,-
JPEG2000 (Du and Fowler, 2007)	880.2	20.889	0.1	-,-
PCA-DCT (Nian et al., 2016)	880.2	27.302	0.1	-,-
PCA+JPEG2000 (Du and Fowler, 2007)	880.2	27.5	0.1	-,-
FPCA+JPEG2000 (Mei et al., 2018)	880.2	-	0.1	-,-
HEVC (Sullivan et al., 2012)	880.2	31	0.1	-,-
RPM (Paul et al., 2016)	880.2	34	0.1	-,-
3D SPECK (Tang and Pearlman, 2006)	880.2	27.1	0.1	-,-
3D DCT (Yadav and Nagmode, 2018)	880.2	33.4	0.1	-,-
3D DWT+SVR (Zikiou et al., 2020)	880.2	28.20	0.1	-,-
WSRC (Ouahioune et al., 2021)	880.2	35	0.1	[illegible]
...R (Rezasoltani and Qureshi, 2023)	880.2	28.954	0.1	[illegible]
...INR (Rezasoltani and Qureshi, 2023)	440.1	24.334	0.06	[illegible]
...mpling (Rezasoltani and Qureshi, 2024)	880.2	37.007	0.1	[illegible]
...sampling (Rezasoltani and Qureshi, 2024)	440.1	24.96	0.06	[illegible]
Meta_learning	**0.8**	**33.6**	[illegible]	[illegible]

2nd best

Comparison (compression times)

Dataset	Method	bpppb	compression time (Sec)	decompression time (Sec)	PSNR ↑
Indian Pines	JPEG+ [Good et al., 1994, Qiao et al., 2014]	0.1	7.353	3.27	27.47
	JPEG2000+ [Du and Fowler, 2007]	0.1	0.1455	0.3115	33.58
	PCA-DCT+ [Nian et al., 2016]	0.1	1.66	0.04	32.28
	ours-32bit	0.1	243.64	0	36.98
	ours-16bit	[illegible]	243.64	0	36.95
	ours-sampling-32	[illegible]	282.08	0.0005	**40.1**
	ours-sampling-16	[illegible]	282.08	0.0005	28.40
	meta-learning	[illegible]	**0.033**	**0.000717**	33.36
Jasper Ridge	JPEG+ [Good et al., 1994, Qiao et al., 2014]	0.1	3.71	1.62	21.13
	JPEG2000+ [Du and Fowler, 2007]	0.1	0.138	0.395	17.49
	PCA-DCT+ [Nian et al., 2016]	0.1	1.029	0.027	26.82
	ours-32bit	0.1	312.38	0.0005	32.54
	ours-16bit	[illegible]	312.38	0.0005	32.51
	ours-sampling-32	[illegible]	75.91	0.0005	**34.77**
	ours-sampling-16	[illegible]	75.91	0.0005	22.07
	meta-learning	[illegible]	**0.025**	**0.0007**	30.87
Pavia University	JPEG+ [Good et al., 1994, Qiao et al., 2014]	0.1	33.86	14.61	20.25
	JPEG2000+ [Du and Fowler, 2007]	0.1	0.408	0.628	17.75
	PCA-DCT+ [Nian et al., 2016]	0.1	6.525	0.235	25.43
	ours-32bit	0.1	780.16	0.0009	34.46
	ours-16bit	[illegible]	780.16	0.0009	34.17
	ours-sampling-32	[illegible]	72.512	0.0004	**38.08**
	ours-sampling-16	[illegible]	72.512	0.0004	27.02
	meta-learning	[illegible]	**0.43**	**0.0006**	35.3
Cuprite	JPEG+ [Good et al., 1994, Qiao et al., 2014]	0.06	101.195	45.02	12.88
	JPEG2000+ [Du and Fowler, 2007]	0.06	1.193	2.476	15.16
	PCA-DCT+ [Nian et al., 2016]	0.06	11.67	0.754	26.75
	ours-32bit	0.06	1565.97	0.001	28.02
	ours-16bit	[illegible]	1565.97	0.001	27.90
	ours-sampling-32	[illegible]	664.87	0.001	**37.27**
	ours-sampling-16	[illegible]	664.87	0.001	24.85
	meta-learning	[illegible]	**1.11**	**0.0007**	24.57

Best

Best

Best

Best

Comparison (decompression times)

Dataset	Method	bppppb	compression time (Sec)	decompression time (Sec)	PSNR ↑
Indian Pines	JPEG+ [Good et al., 1994, Qiao et al., 2014]	0.1	7.353	3.27	27.47
	JPEG2000+ [Du and Fowler, 2007]	0.1	0.1455	0.3115	33.58
	PCA-DCT+ [Nian et al., 2016]	0.1	1.66	0.04	32.28
	ours-32bit	0.1	243.64	0	36.98
	ours-16bit	0.05	243.64	0	36
	ours-sampling-32bit	0.1	282.08	0.0005	**40**
	ours-sampling-16bit	0.05	282.08	0.0005	28
	meta-learning	6.9e-6	**0.033**	**0.000717**	
Jasper Ridge	JPEG+ [Good et al., 1994, Qiao et al., 2014]	0.1	3.71	1.62	21.13
	JPEG2000+ [Du and Fowler, 2007]	0.1	0.138	0.395	17.49
	PCA-DCT+ [Nian et al., 2016]	0.1	1.029	0.027	26.82
	ours-32bit	0.1	312.38	0.0005	32.54
	ours-16bit	0.06	312.38	0.0005	32
	ours-sampling-32bit	0.1	75.91	0.0005	**34**
	ours-sampling-16bit	0.06	75.91	0.0005	22
	meta-learning	1.4e-5	**0.025**	**0.0007**	
Pavia University	JPEG+ [Good et al., 1994, Qiao et al., 2014]	0.1	33.86	14.61	20.25
	JPEG2000+ [Du and Fowler, 2007]	0.1	0.408	0.628	17.75
	PCA-DCT+ [Nian et al., 2016]	0.1	6.525	0.235	25.43
	ours-32bit	0.1	780.16	0.0009	34.46
	ours-16bit	0.05	780.16	0.0009	34
	ours-sampling-32bit	0.1	72.512	0.0004	**38**
	ours-sampling-16bit	0.05	72.512	0.0004	27
	meta-learning	0.003	**0.43**	**0.0006**	
Cuprite	JPEG+ [Good et al., 1994, Qiao et al., 2014]	0.06	101.195	45.02	12.88
	JPEG2000+ [Du and Fowler, 2007]	0.06	1.193	2.476	15.16
	PCA-DCT+ [Nian et al., 2016]	0.06	11.67	0.754	26.75
	ours-32bit	0.06	1565.97	0.001	28.02
	ours-16bit	0.03	1565.97	0.001	27
	ours-sampling-32bit	0.06	664.87	0.001	**37**
	ours-sampling-16bit	0.03	664.87	0.001	24
	meta-learning	4.5e-7	**1.11**	**0.0007**	

Best

Best

Best

Best

Reducing compression times

Proposal: Exploit spatial and spectral similarities between hyperspectral images using *meta learning* to achieve faster compression

Question 3: Is it possible to achieve faster compression at acceptable PSNR using *meta learning*?

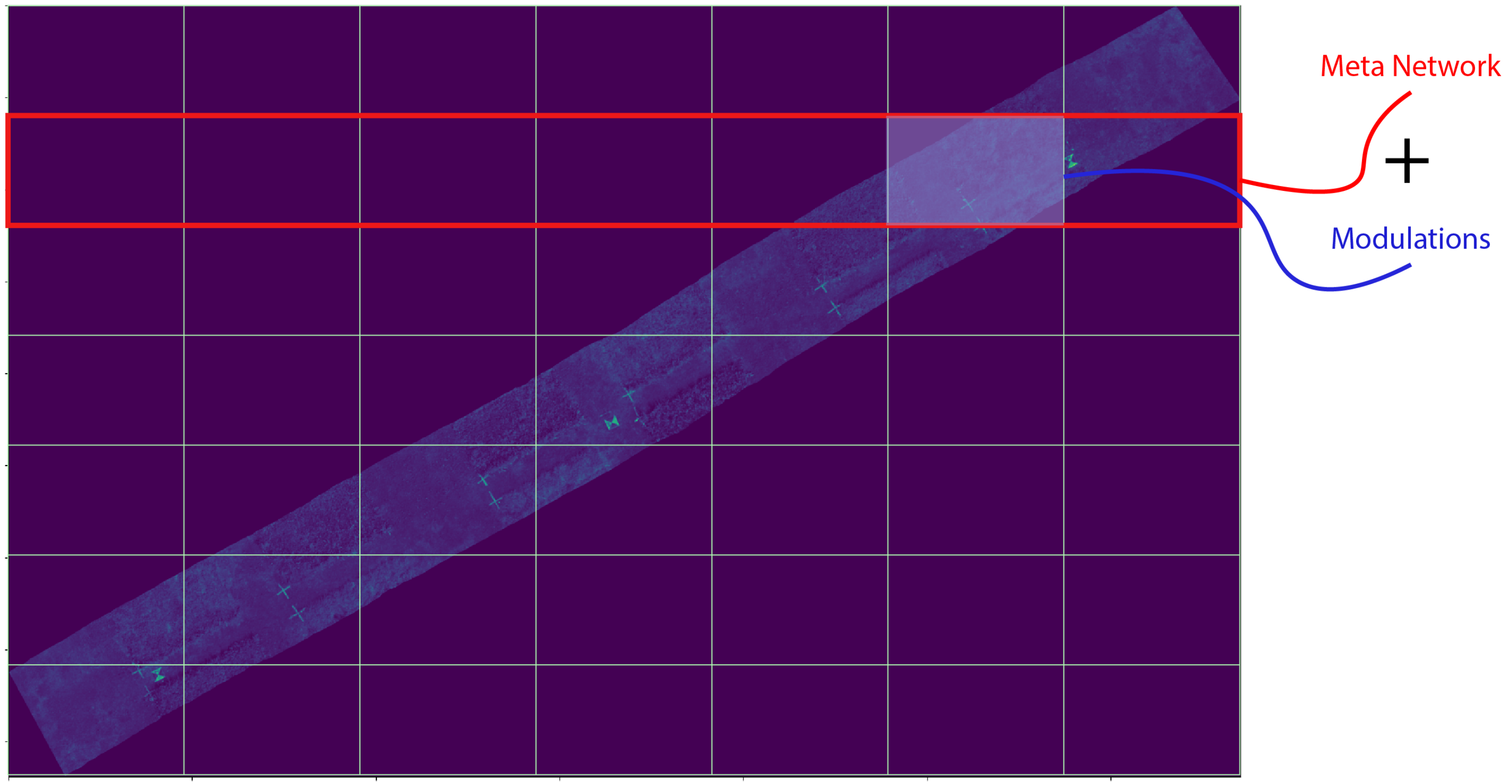
YES

Proof of Concept

Can we use implicit neural represent to compress “large” hyperspectral image?

Proof of Concept

Can we use implicit neural represent to compress “large” hyperspectral image?

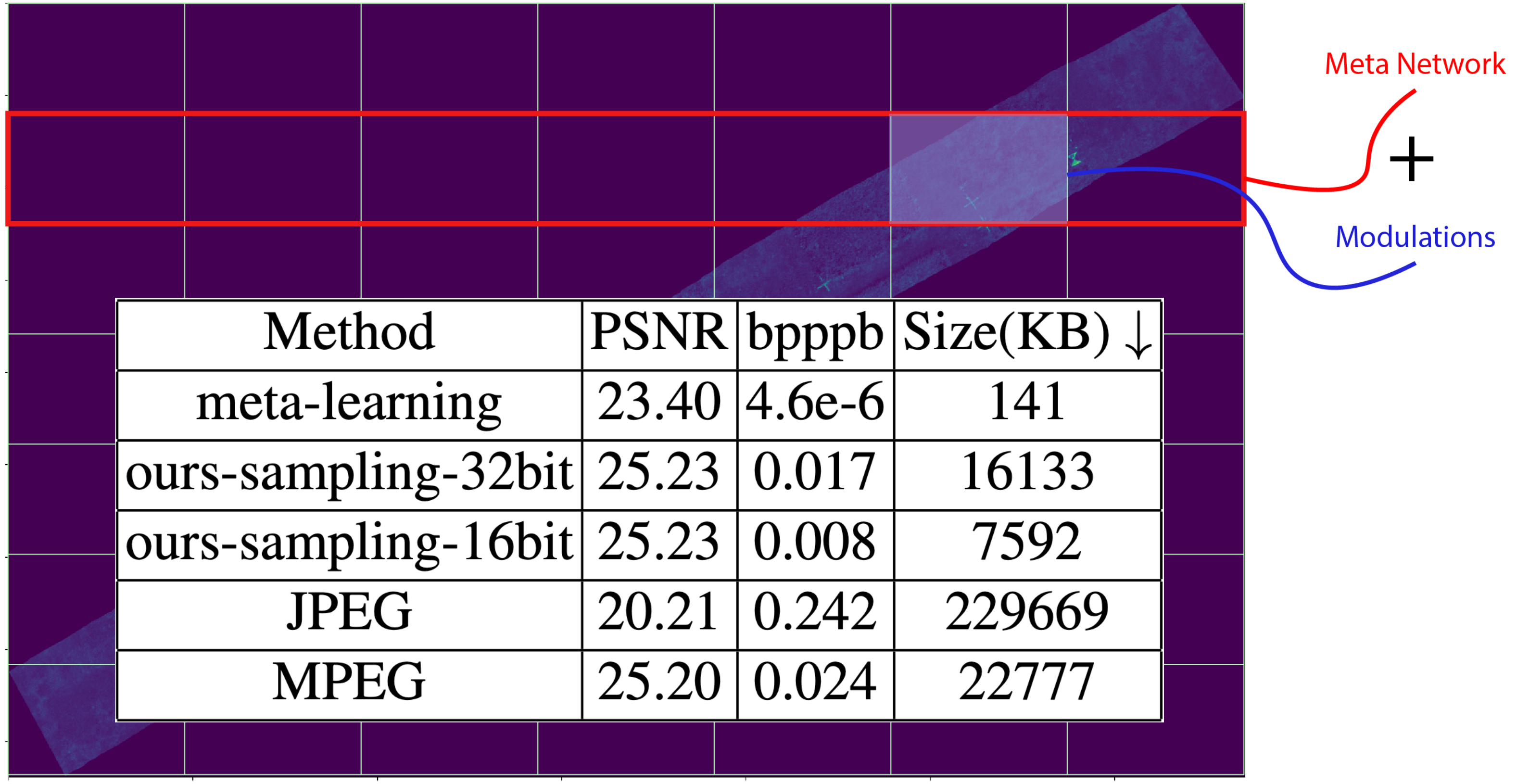


Train 7 meta networks, on per row

Modulations capture structure within tiles in each rows

Proof of Concept

Can we use implicit neural represent to compress "large" hyperspectral image?



Method	PSNR	bpppb	Size(KB) ↓
meta-learning	23.40	4.6e-6	141
ours-sampling-32bit	25.23	0.017	16133
ours-sampling-16bit	25.23	0.008	7592
JPEG	20.21	0.242	229669
MPEG	25.20	0.024	22777

Train 7 meta networks, on per row

Modulations capture structure within tiles in each rows

*This table fixes an error in the thesis

Proof of Concept

Can we use implicit neural represent to compress "large" hyperspectral image?

YES

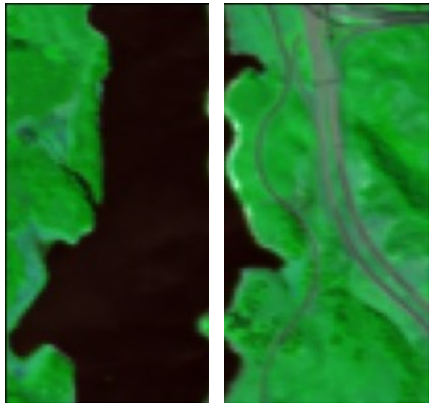
Compressed Size

141 KB

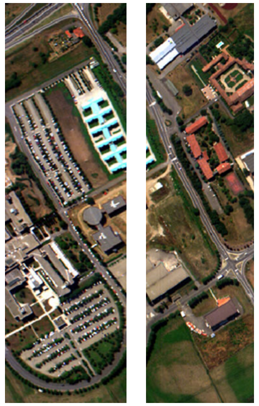
Task-Aware Compression

Is it possible to compress regions of an image differentially?

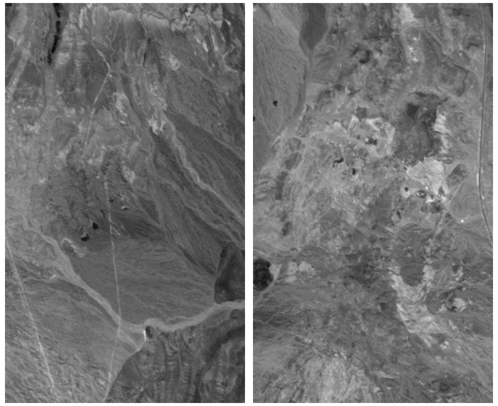
Indian Pine
PSNR: 33.47
PSNR left side: 33.82
PSNR right side: 33.15



Jasper Ridge
PSNR: 28.15
PSNR left side: 29.82
PSNR right side: 26.94



Pavia University
PSNR: 30.54
PSNR left side: 29.40
PSNR right side: 32.11



Cuprite
PSNR: 23.66
PSNR left side: 23.66
PSNR right side: 23.67

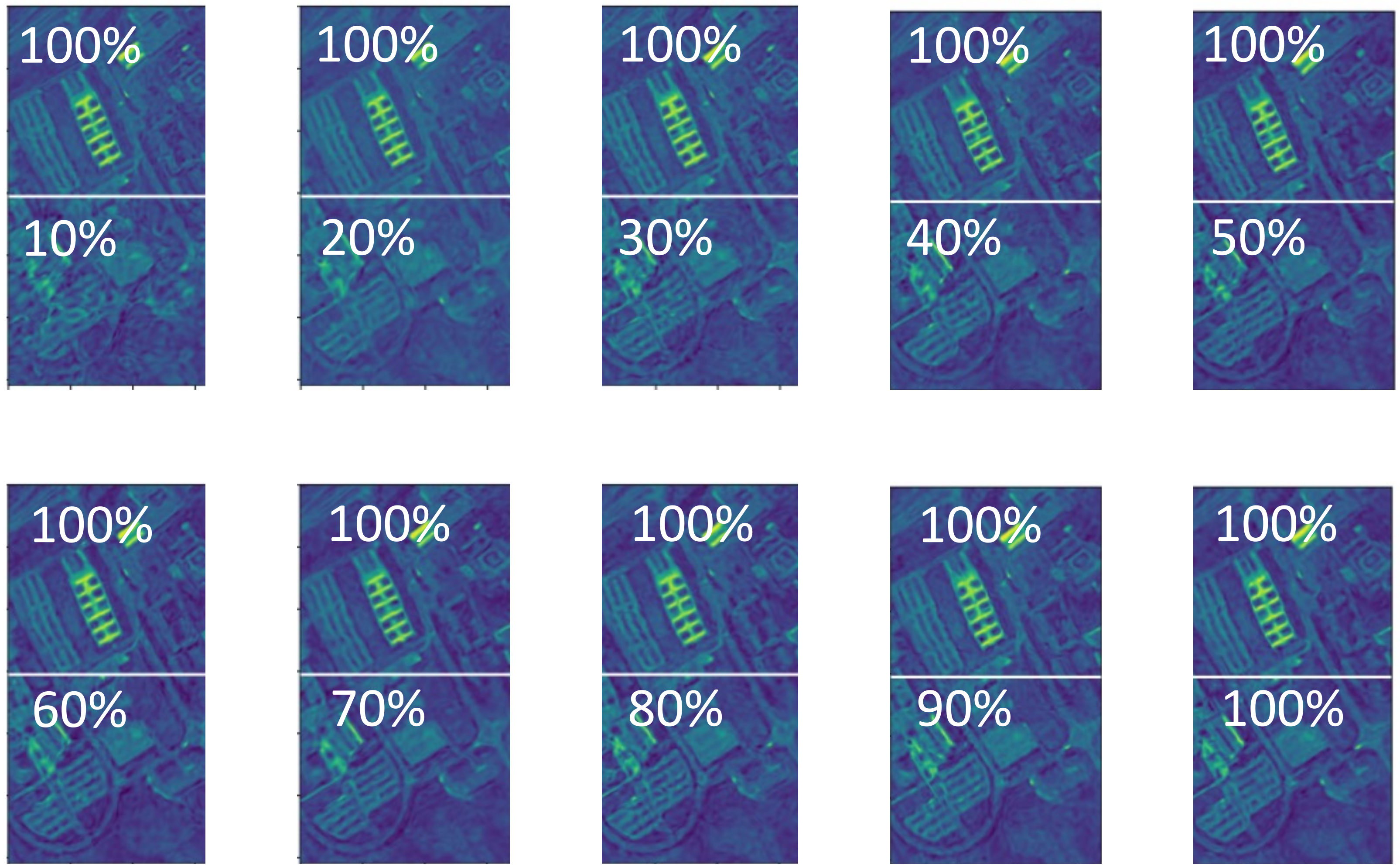
Region-Specific Compression on Pavia Dataset

Slice 2 Sampling Rate	Slice 1 PSNR	Slice 2 PSNR
10	26.72	23.84
20	26.60	24.72
30	26.51	25.22
40	26.49	25.54
50	26.23	25.69
60	26.17	25.79
70	26.25	25.95
80	26.06	26.12
90	26.06	26.15
100	25.99	26.21

Region-Specific Compression

Pavia University

Top and bottom slices sampled at different rates



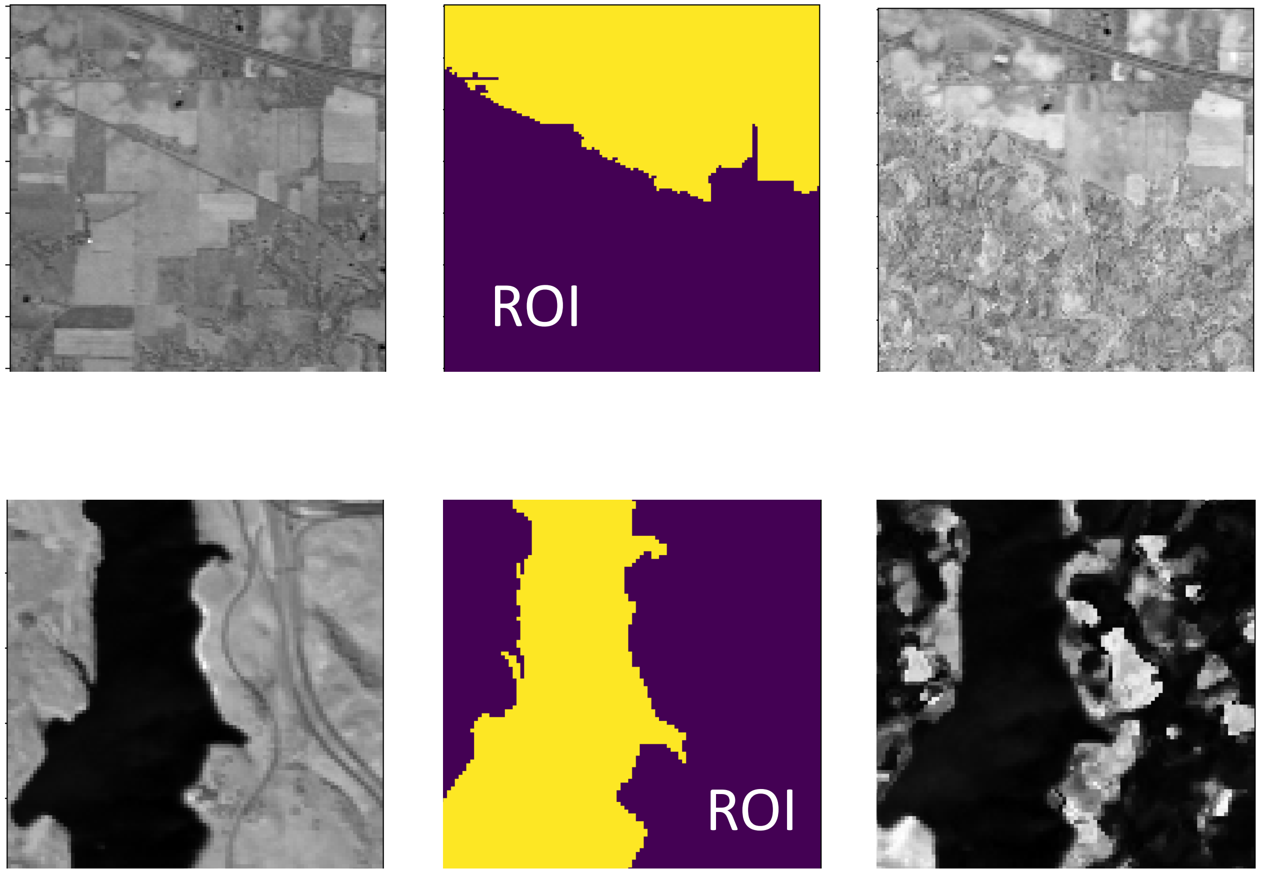
Region-Specific Compression

- Regions of Interest (ROI)
  - K-Means
    - Clusters regions based upon spectral similarity
  - UNet
    - Uses deep learning to perform object-level segmentation

K-Means for ROI: An illustration

Indian Pines

Jasper Ridge

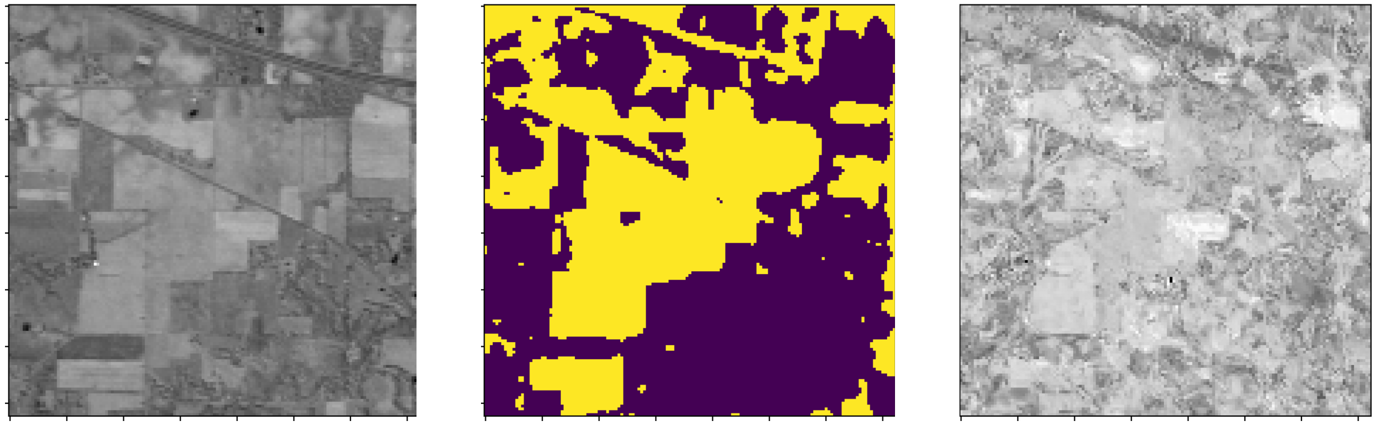


Results K-means

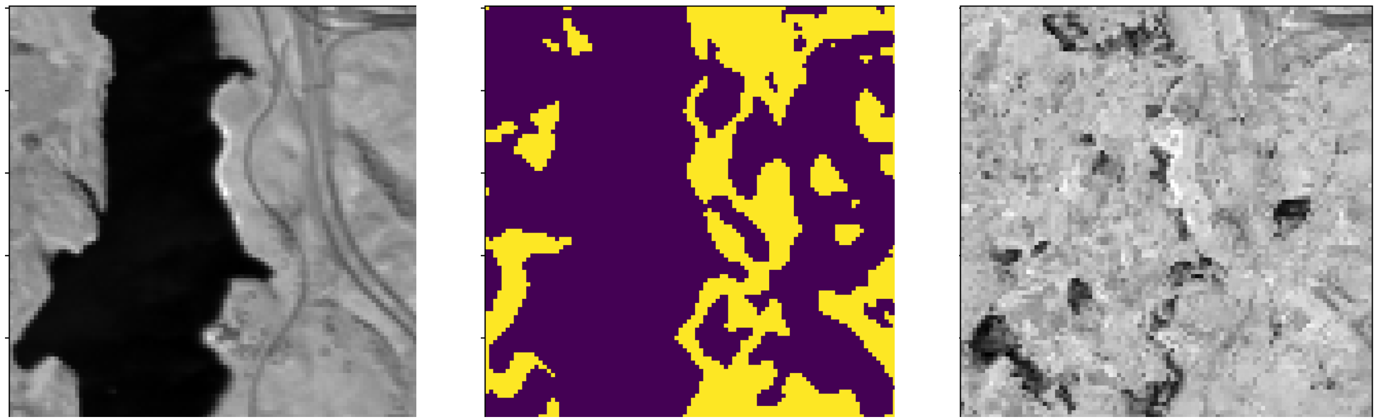
Dataset	PSNR-ROI	PSNR	bpppb-ROI	bpppb	Compressed-size (KB)	Original-size (KB)
Indian Pines	42.28	25.94	0.072	0.28	332.6	9251
Jasper Ridge	38.21	13.62	1.54	0.59	333.5	4800
Pavia University	37.67	20.40	0.14	0.05	304	42724
Cuprite	37.54	22.03	0.08	0.01	333.5	140836
Large dataset	38.05	16.89	7.06	0.008	344.8	2.82e+7

UNet for ROI: An illustration

Indian Pines



Jasper Ridge



ROI are shown in Yellow

Results U-net

Dataset	PSNR-ROI	PSNR	bpppb-ROI	bpppb	Compressed-size(KB)	Original-size(KB)
Indian Pines	35.07	25.46	0.66	0.28	332.6	9251
Jasper Ridge	26.43	13.79	2.11	0.59	333.5	4800
Pavia University	32.39	21.28	0.140	0.05	304.0	42724
Cuprite	26.21	23.75	0.040	0.018	333.5	140836
Large dataset	36.009	27.82	0.180	0.008	344.8	2.82e+7

Task-Aware Compression

Is it possible to compress regions of an image differentially?

YES

Thesis Questions

- Is it possible to achieve high compression rates while maintaining acceptable quality when using implicit neural representations?
- Is it possible to achieve high compression rates while maintaining acceptable quality when using *sampling*?
- Is it possible to achieve faster compression at acceptable PSNR using *meta learning*?
- Can we use implicit neural represent to compress “large” hyperspectral image?
- Is it possible to compress regions of an image differentially?

Questions

- Is it possible to achieve high compression rates while maintaining acceptable quality when using implicit neural representations?
- Is it p … ining acce
- Is it p … NR using *meta*
- Can w … hyper
- Is it possible to compress regions of an image differentially?

The results suggest that the answer to all these questions is a YES!

Contributions

- Explored the use of implicit neural representations for hyperspectral image compression
  - Sampling
  - Meta learning
  - Managing large-scale images
  - Differential compression
- Evaluated on standard benchmarks against state-of-the-art schemes, posting competitive performance

Publications

- **Hyperspectral Image Compression Using Implicit Neural Representations**. Rezasoltani, S.; and Qureshi, F. In *Proc. 20th Conference on Robots and Vision (CRV23)*, pages 8pp, Montreal, Jun 2023.
- **Hyperspectral Image Compression Using Sampling and Implicit Neural Representations**. Rezasoltani, S.; and Qureshi, F. Z. *IEEE Transactions on Geoscience and Remote Sensing*, 63: 12pp. December 2024.
  (top journal in the field of remote sensing, impact factor: 7.5)
- **Hyperspectral Image Compression Using Implicit Neural Representation and Meta-Learned Based Network**. Rezasoltani, S.; and Qureshi, F. Z. In *Proc. 14th International Conference on Pattern Recognition Applications and Methods*, pages 9pp, Porto, February 2025. (Honorable Mention)
- **Meta-Learned Implicit Neural Representations for Scalable and Fast Hyperspectral Image Compression**. Rezasoltani, S.; and Qureshi, F.Z. Lecture Notes in Computer Science – ICPRAM 2025 Selected Papers, Springer, pages 18pp (In review)

Limitations

- Theoretical understanding of the limits of implicit neural representations from an information theoretic perspective: model capacity vs. compression quality
  - We have side-stepped this issue in this thesis via “architecture search”
- More rigorous evaluation of model performance in compression large hyperspectral images
- Evaluation of the proposed technique on more benchmarks
  - We have evaluated the model on the benchmarks that are currently used in the literature, but clearly it is desirable to evaluate to model on a wider set of benchmarks

Future Work

- Apply this framework to multispectral and medical imaging types to improve storage and diagnostics.
- Enable incremental learning and make it compatible with edge devices for real-time use in satellites and autonomous vehicles.
- Build privacy-preserving frameworks and establish guidelines for using sensitive data responsibly.

References

- S. Rezasoltani and F. Qureshi, “Hyperspectral Image Compression Using Implicit Neural Representation and Meta-Learned Based Network,” in Proceedings of the 14th International Conference on Pattern Recognition Applications and Methods (ICPRAM), pp. 23–31, Feb. 2025.
- S. Rezasoltani and F. Qureshi, “Hyperspectral Image Compression Using Sampling and Implicit Neural Representations,” IEEE Transactions on Geoscience and Remote Sensing, Dec. 2024.
- S. Rezasoltani and F. Qureshi, “Hyperspectral Image Compression Using Implicit Neural Representations,” in 2023 20th Conference on Computer and Robot Vision (CRV), pp. 31–38, May 2023.
- J. Zhang, Y. Zhang, X. Cai, and L. Xie, “Three-stages hyperspectral image compression sensing with band selection.” CMES-Computer Modeling in Engineering & Sciences, vol. 134, no. 1, 2023.
- Y. Guo, Y. Chong, and S. Pan, “Hyperspectral image compression via cross-channel contrastive learning,” IEEE Transactions on Geoscience and Remote Sensing, 2023.
- S. Bajpai, N. R. Kidwai, H. V. Singh, and A. K. Singh, “A low complexity hyperspectral image compression through 3d set partitioned embedded zero block coding,” Multimedia Tools and Applications, pp. 1–32, 2022.
- L. Wang and W. Wang, “Hyperspectral image compressed processing: Evolutionary multi-objective optimization sparse decomposition,” Plos one, vol. 17, no. 4, p. e0267754, 2022.
- Y. Guo, Y. Tao, Y. Chong, S. Pan, and M. Liu, “Edge-guided hyperspectral image compression with interactive dual attention,” IEEE Transactions on Geoscience and Remote Sensing, vol. 61, pp. 1–17, 2022.
- B. Sujitha, V. S. Parvathy, E. L. Lydia, P. Rani, Z. Polkowski, and K. Shankar, “Optimal deep learning based image compression technique for data transmission on industrial internet of things applications,” Trans- actions on Emerging Telecommunications Technologies, vol. 32, no. 7, p.e3976, 2021.
- M. P. Uddin, M. A. Mamun, and M. A. Hossain, “Pca-based feature reduction for hyperspectral remote sensing image classification,” IETE Technical Review, vol. 38, no. 4, pp. 377–396, 2021.

Previous Research

Method	Datasets	Reference
Transform-based		
Sampling design and uncertainty based on spatial variability of spectral variables for mapping vegetation cover	Indian Pines, Pavia	[Wang et al., 2005]
Hyperspectral image compression: adapting SPIHT and EZW to anisotropic 3-D wavelet coding	AVIRIS	[Christophe et al., 2008]
Hyperspectral image compression based on tucker decomposition and discrete cosine transform	AVIRIS	[Karami et al., 2010]
Lossless hyperspectral image compression using wavelet transform based spectral decorrelation	AVIRIS	[Toreyın et al., 2015]
Lossy compression of Landsat multispectral images	Landsat	[Kozhemiakin et al., 2016]
ROI-based on-board compression for hyperspectral remote sensing images on GPU	AVIRIS	[Giordano et al., 2017]
A new algorithm for the on-board compression of hyperspectral images	Indian Pines, Pavia	[Guerra et al., 2018]
Hyperspectral image compression using vector quantization, PCA, and JPEG2000	Cuprite	[Bascones et al., 2018]
PCA-based feature reduction for hyperspectral remote sensing image classification	Indian Pines	[Uddin et al., 2021]
Three-stages hyperspectral image compression sensing with band selection	Indian Pines, Pavia	[Cai et al., 2022]
Learning-based		
Hyperspectral image compression based on online learning spectral features dictionary	AVIRIS	[Jifara et al., 2017]
LSTM based adaptive filtering for reduced prediction errors of hyperspectral images	Indian Pines, Pavia	[Jiang et al., 2018]
Onboard hyperspectral image compression using compressed sensing and deep learning	Pavia	[Kumar et al., 2018]
Large-scale hyperspectral image compression via sparse representations based on online learning	AVIRIS	[Ulku et al., 2018]
The linear prediction vector quantization for hyperspectral image compression	AVIRIS	[Li et al., 2019]
Hyperspectral image compression and super-resolution using tensor decomposition learning	EUROSAT	[Aidini et al., 2019]
Auto encoder-based dimensionality reduction and classification using convolutional neural networks for hyperspectral images	Pavia	[Ramamurthy et al., 2020]
Optimal deep learning-based image compression technique for data transmission on industrial Internet of things applications	SIPI	[Sujitha et al., 2021]
Edge-guided hyperspectral image compression with interactive dual attention	Pavia, Cave	[Guo et al., 2022]
Hyperspectral image compressed processing: Evolutionary multi-objective optimization sparse decomposition	Cuprite, Indian Pines, Pavia	[Wang et al., 2022]
Hyperspectral image compression via cross-channel contrastive learning	Pavia, Cave	[Guo et al., 2023]

Previous Research

Method	Datasets	Reference
Transform-based		
Sampling design and uncertainty based on spatial variability of spectral variables for mapping vegetation cover	Indian Pines, Pavia	[Wang et al., 2005]
Hyperspectral image compression: adapting SPIHT and EZW to anisotropic 3-D wavelet coding	AVIRIS	[Christophe et al., 2008]
Hyperspectral image compression based on tucker decomposition and discrete cosine transform	AVIRIS	[Karami et al., 2010]
Lossless hyperspectral image compression using wavelet transform based spectral decorrelation	AVIRIS	[Toreyın et al., 2015]
Lossy compression of Landsat multispectral images	Landsat	[Kozhemiakin et al., 2016]
ROI-based on-board compression for hyperspectral remote sensing images on GPU	AVIRIS	[Giordano et al., 2017]
A new algorithm for the on-board compression of hyperspectral images	Indian Pines, Pavia	[Guerra et al., 2018]
Hyperspectral image compression using vector quantization, PCA, and JPEG2000	Cuprite	[Bascones et al., 2018]
PCA-based feature reduction for hyperspectral remote sensing image classification	Indian Pines	[Uddin et al., 2021]
Three-stages hyperspectral image compression sensing with band selection	Indian Pines, Pavia	[Cai et al., 2022]
Learning-based		
Hyperspectral image compression based on online learning spectral features dictionary	AVIRIS	[Jifara et al., 2017]
LSTM based adaptive filtering for reduced prediction errors of hyperspectral images	Indian Pines, Pavia	[Jiang et al., 2018]
Onboard hyperspectral image compression using compressed sensing and deep learning	Pavia	[Kumar et al., 2018]
Large-scale hyperspectral image compression via sparse representations based on online learning	AVIRIS	[Ulku et al., 2018]
The linear prediction vector quantization for hyperspectral image compression	AVIRIS	[Li et al., 2019]
Hyperspectral image compression and super-resolution using tensor decomposition learning	EUROSAT	[Aidini et al., 2019]
Auto encoder-based dimensionality reduction and classification using convolutional neural networks for hyperspectral images	Pavia	[Ramamurthy et al., 2020]
Optimal deep learning-based image compression technique for data transmission on industrial Internet of things applications	SIPI	[Sujitha et al., 2021]
Edge-guided hyperspectral image compression with interactive dual attention	Pavia, Cave	[Guo et al., 2022]
Hyperspectral image compressed processing: Evolutionary multi-objective optimization sparse decomposition	Cuprite, Indian Pines, Pavia	[Wang et al., 2022]
Hyperspectral image compression via cross-channel contrastive learning	Pavia, Cave	[Guo et al., 2023]

Metrics

- Peak Signal-to-Noise Ratio (PSNR)
- The PSNR measures the proximity of the original image to its reconstruction

$$PSNR = 10\ \log_{10}\left(\frac{R^2}{MSE}\right)$$

$$MSE = \sum_{i} \frac{\left|I[i] - \tilde{I}[i]\right|^2}{i}$$

Metrics

- Structure similarity (SSIM)
- The SSIM measures the visual quality of the reconstructed image

$$SSIM(x, y) = \frac{(2\,\mu_x\,\mu_y + C_1)\,(2\,\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}$$

Metrics

- The number of bits-per-pixel-per-band (bpppb): captures the level of compression achieved by a model
- Lower values of bpppb indicate higher compression rates
- The parameter bpppb is calculated as follows:

$$bpppb = \frac{\#\text{parameters} \times (\text{bits per parameter})}{(\text{pixels per band}) \times \#\text{bands}}$$

Implicit Neural Representations (INRs)

- Represent an image by overfitting a neural network to it
  - Parameters of the neural network serve as the compact representation of the image
  - Use this image representation as the compressed version of the image
  - Reconstruct the original image by evaluating the neural network at all pixel locations