Learned Compression for Compressed Learning

Dan Jacobellis, Neeraja J. Yadwadkar

TEXAS
The University of Texas at Austin

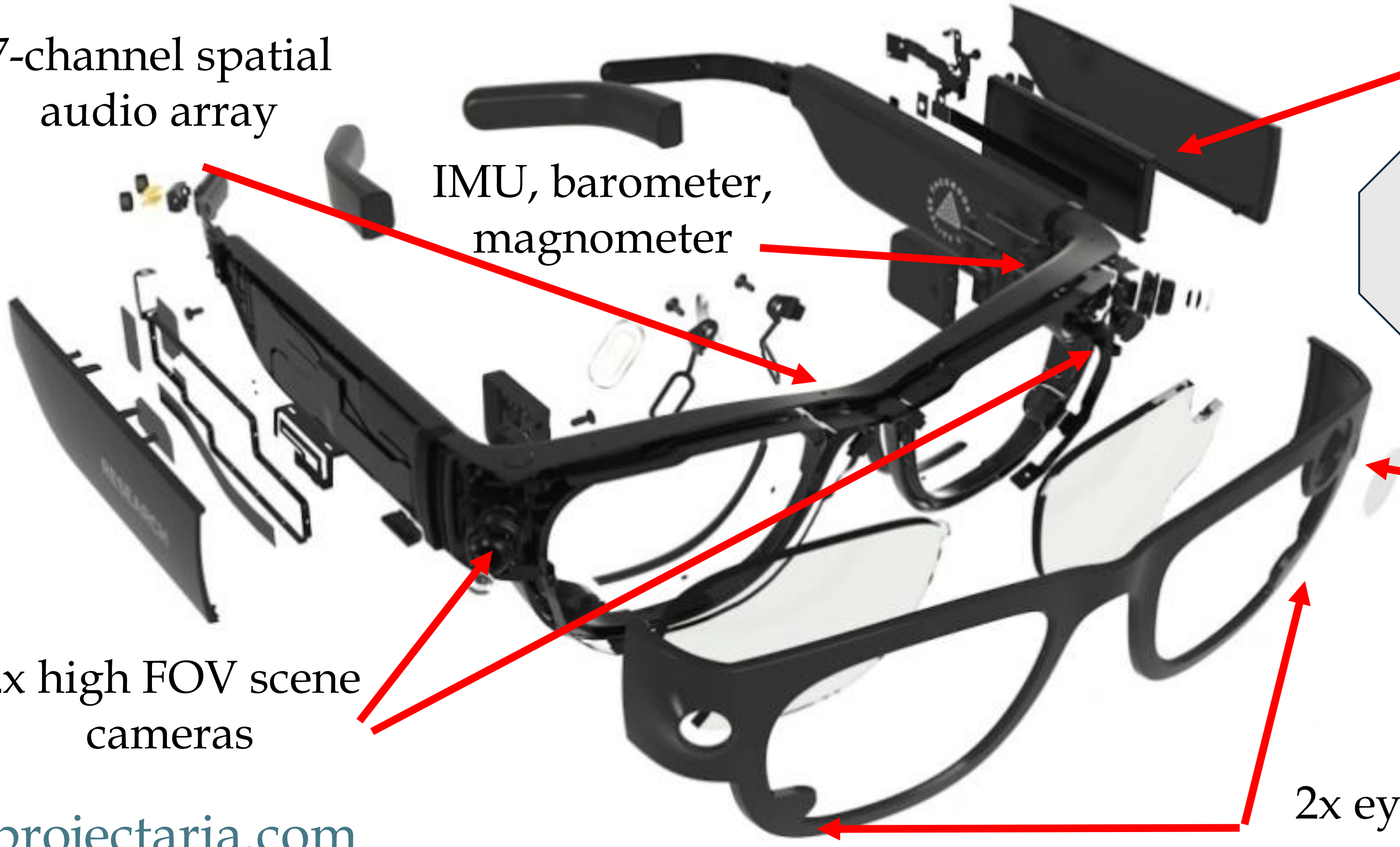
UT-SysML
Systems & Machine Learning

IEEE Data Compression Conference 2025

Compression for mobile and remote sensing

Mobile, remote, and wearable sensors produce constant streams of **high resolution** signals

Sensor efficiency is increasing, while ML models get more expensive



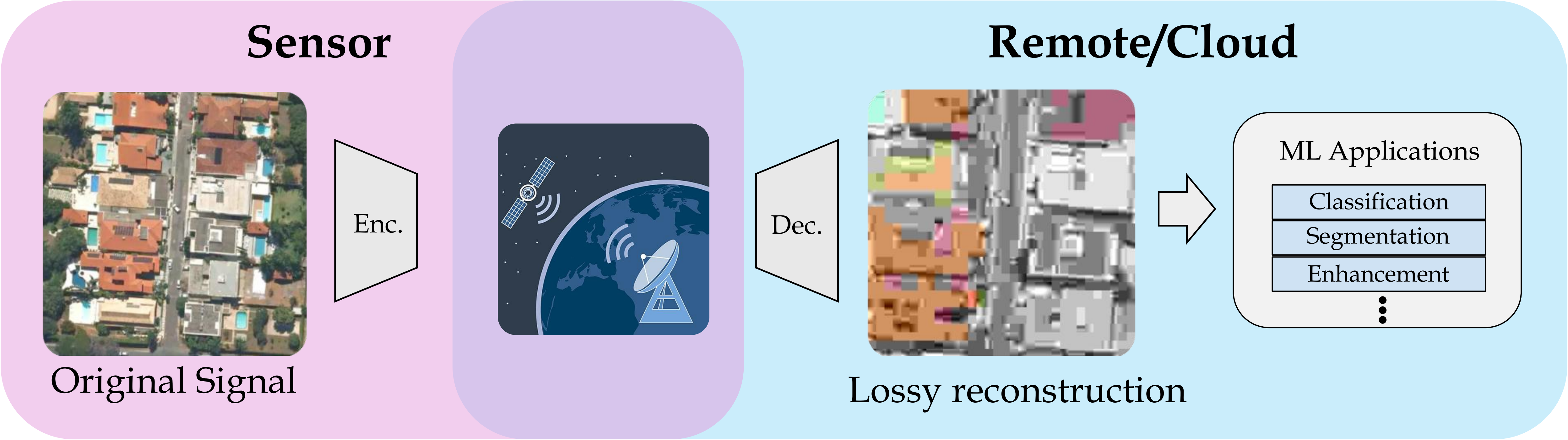
projectaria.com

Compression for mobile and remote sensing

Mobile, remote, and wearable sensors produce constant streams of **high resolution** signals

Sensor efficiency is increasing, while ML models get more expensive

Solution: divide computation between sensor and cloud



Demands high compression ratio

Degrades accuracy

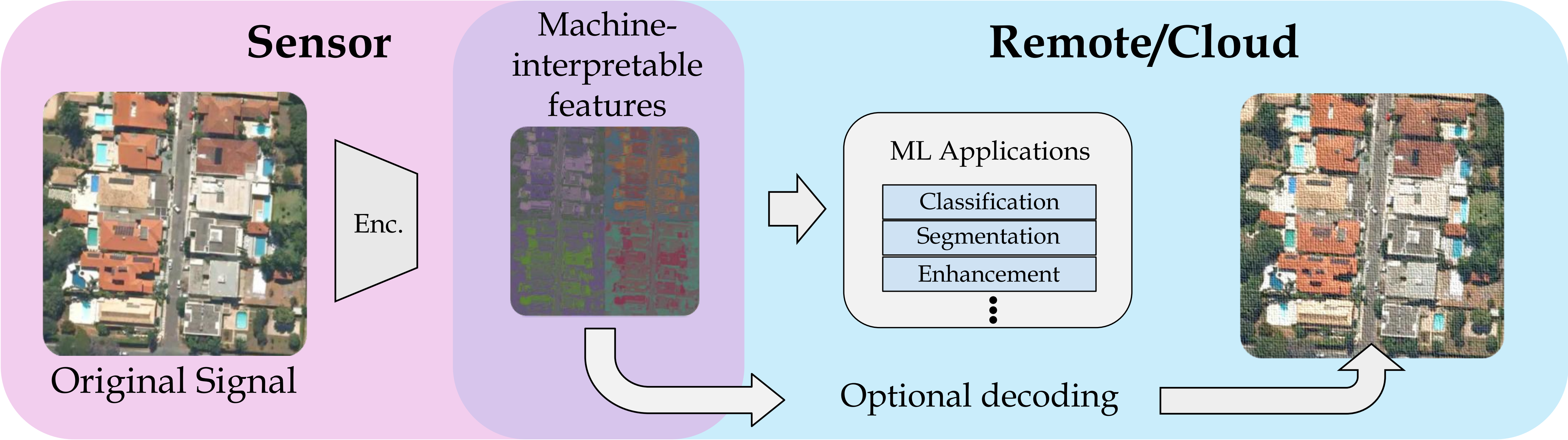
Adds decoding overhead

Machine-oriented compression

Mobile, remote, and wearable sensors produce constant streams of **high resolution** signals

Sensor efficiency is increasing, while ML models get more expensive

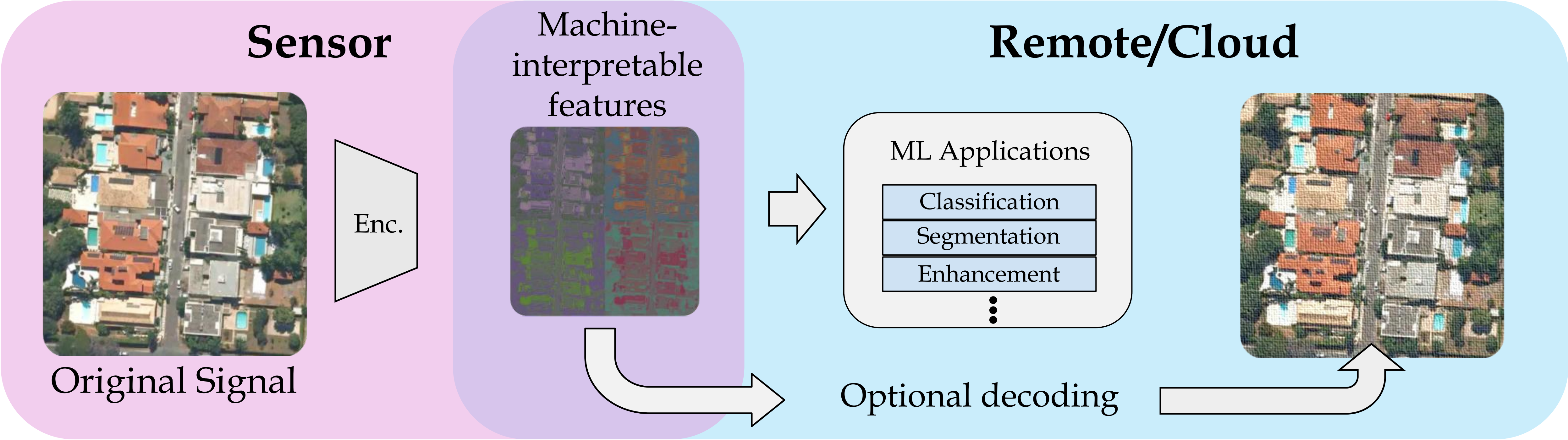
Solution: divide computation between sensor and cloud



Less bandwidth **Enhanced accuracy** **More efficient ML**

Machine-oriented compression

What are ideal characteristics of the compression system?



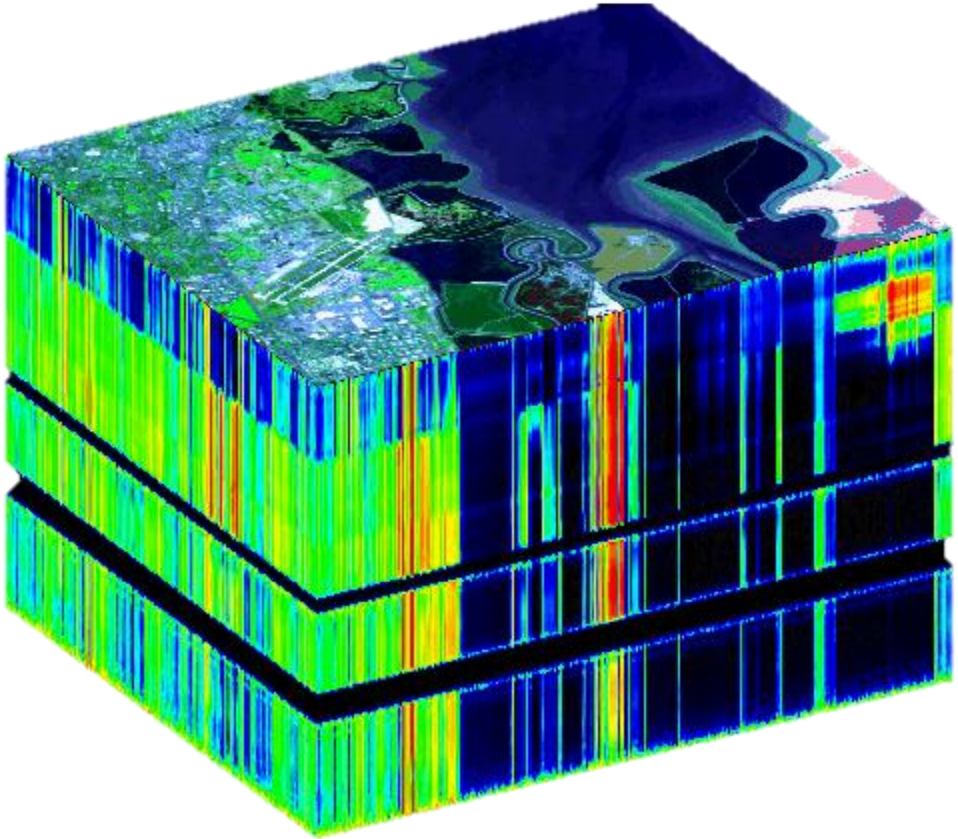
Less bandwidth **Enhanced accuracy** **More efficient ML**

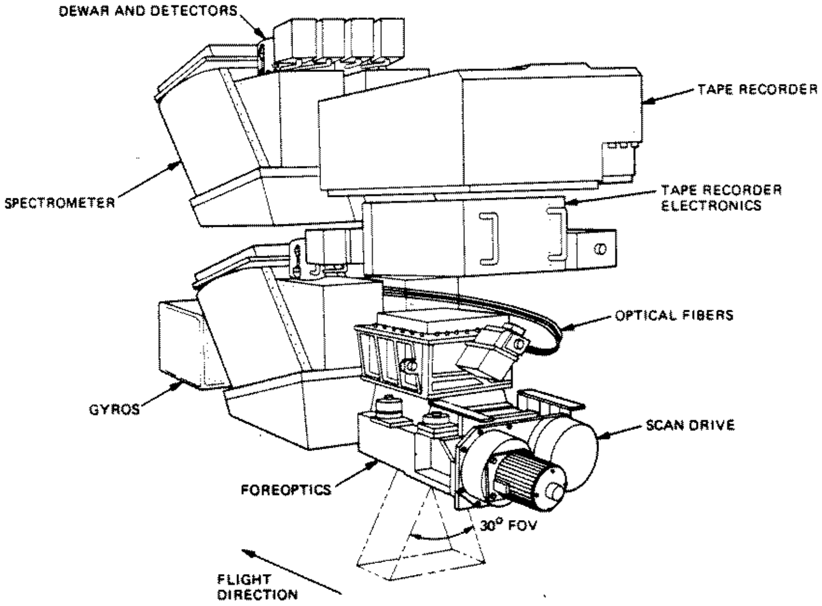
Machine-oriented compression

What are ideal characteristics of the compression system?

- **Support many modalities**

- **Hyperspectral**



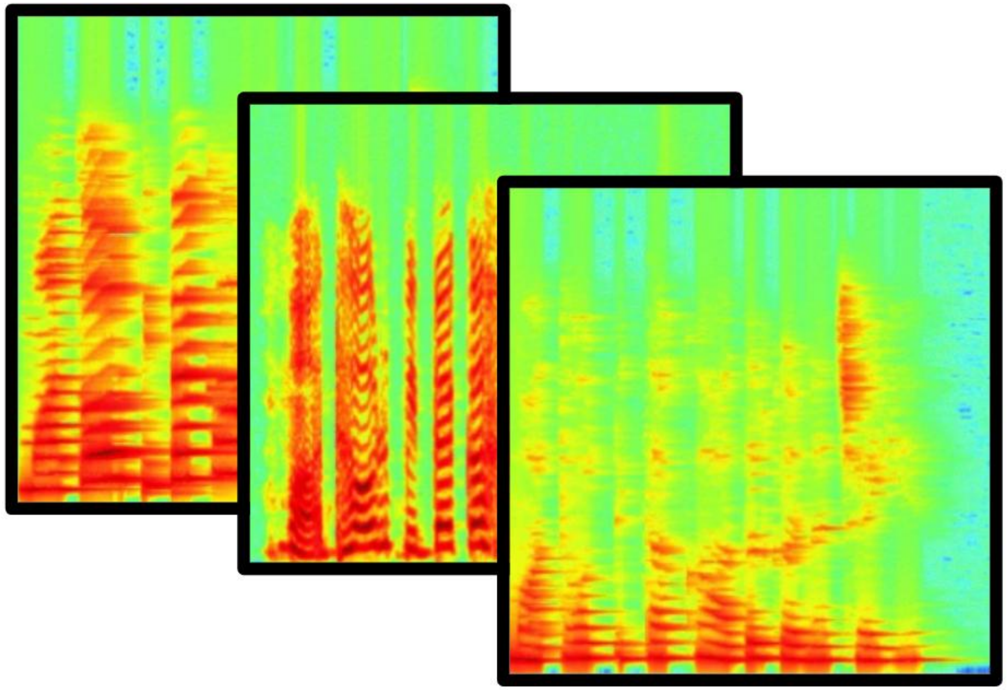


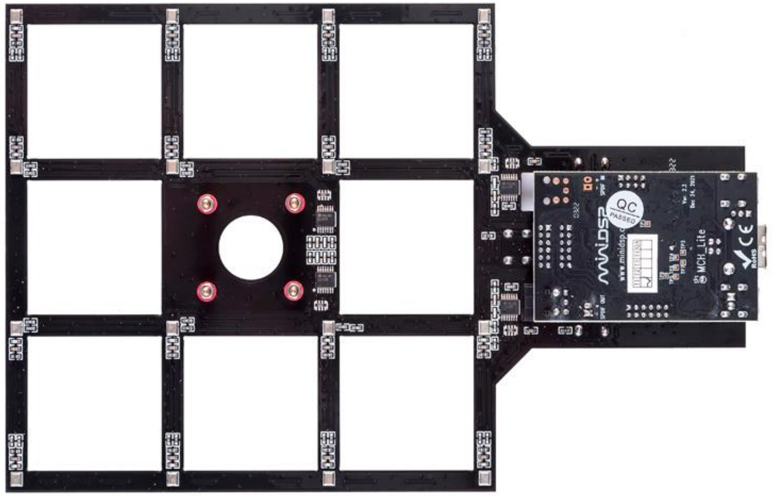
Machine-oriented compression

What are ideal characteristics of the compression system?

- **Support many modalities**

- Hyperspectral
- **Spatial Audio**



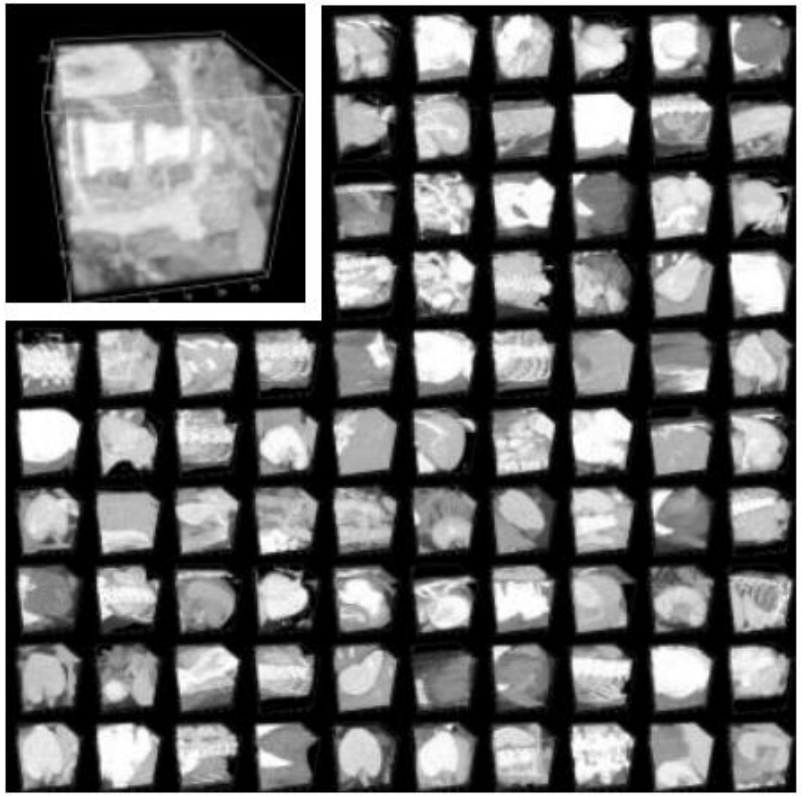


Machine-oriented compression

What are ideal characteristics of the compression system?

- **Support many modalities**

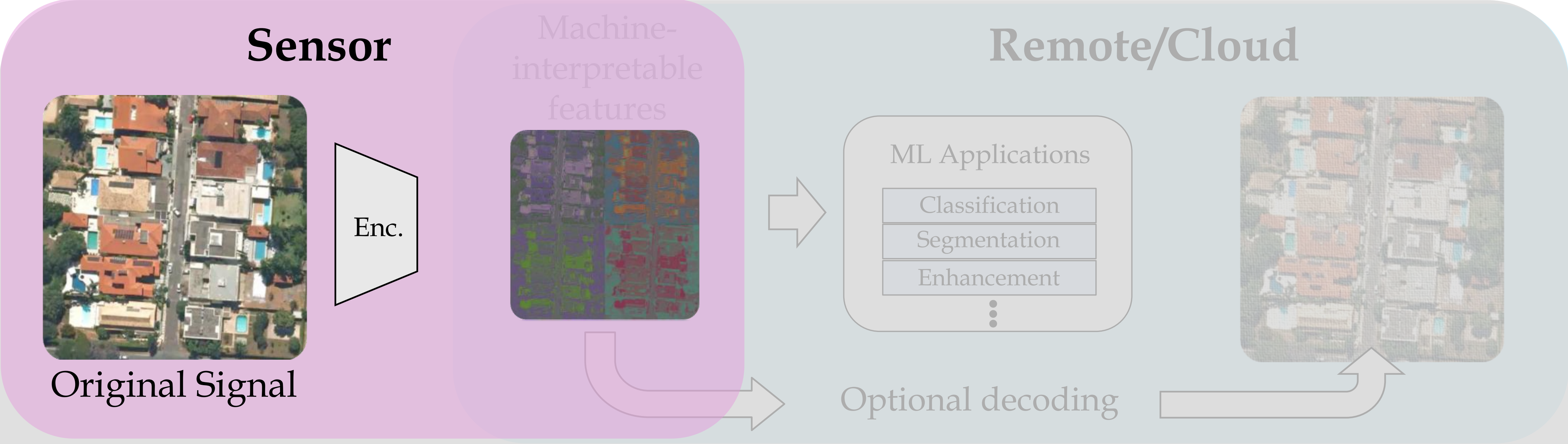
- Hyperspectral
- Spatial Audio
- **3D Computed Tomograpghy**



Machine-oriented compression

What are ideal characteristics of the compression system?

- Support many modalities
- **Allow efficient encoding**



Less bandwidth **Enhanced accuracy** **More efficient ML**

Machine-oriented compression

What are ideal characteristics of the compression system?

- Support many modalities
- Allow efficient encoding
- **Preserve details**

Generative models synthesize details
For recognition, we must **preserve** details

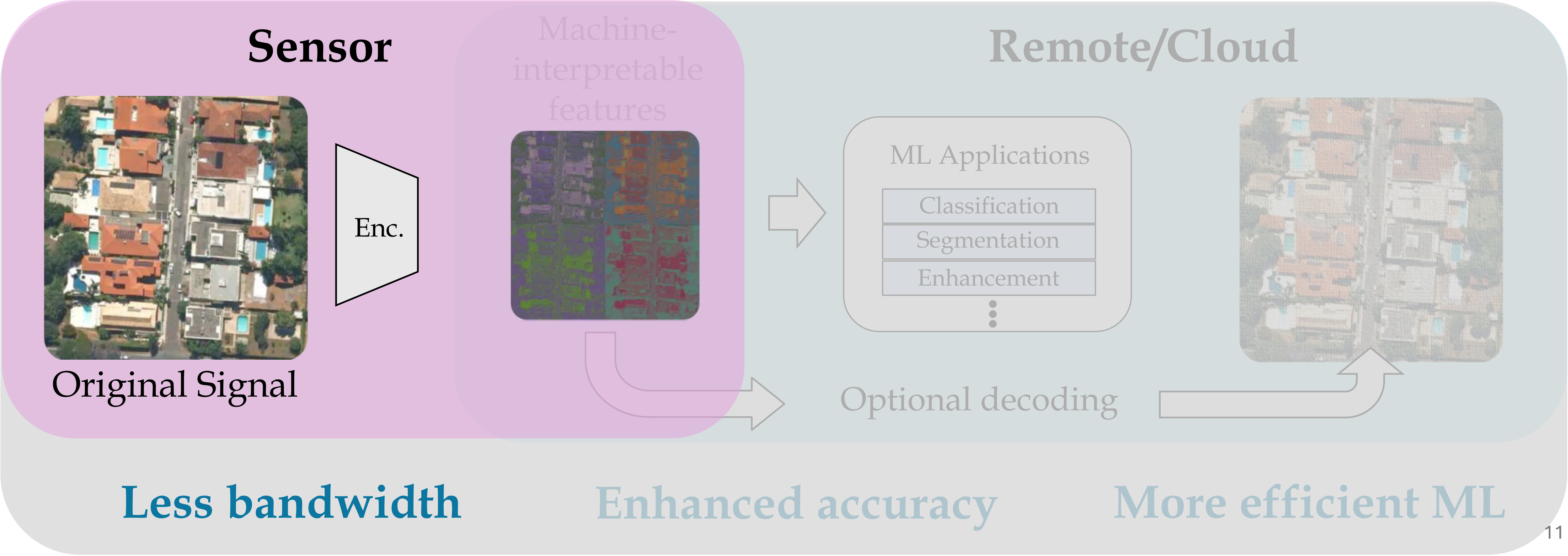
Original

Stable Diff. VAE

Machine-oriented compression

What are ideal characteristics of the compression system?

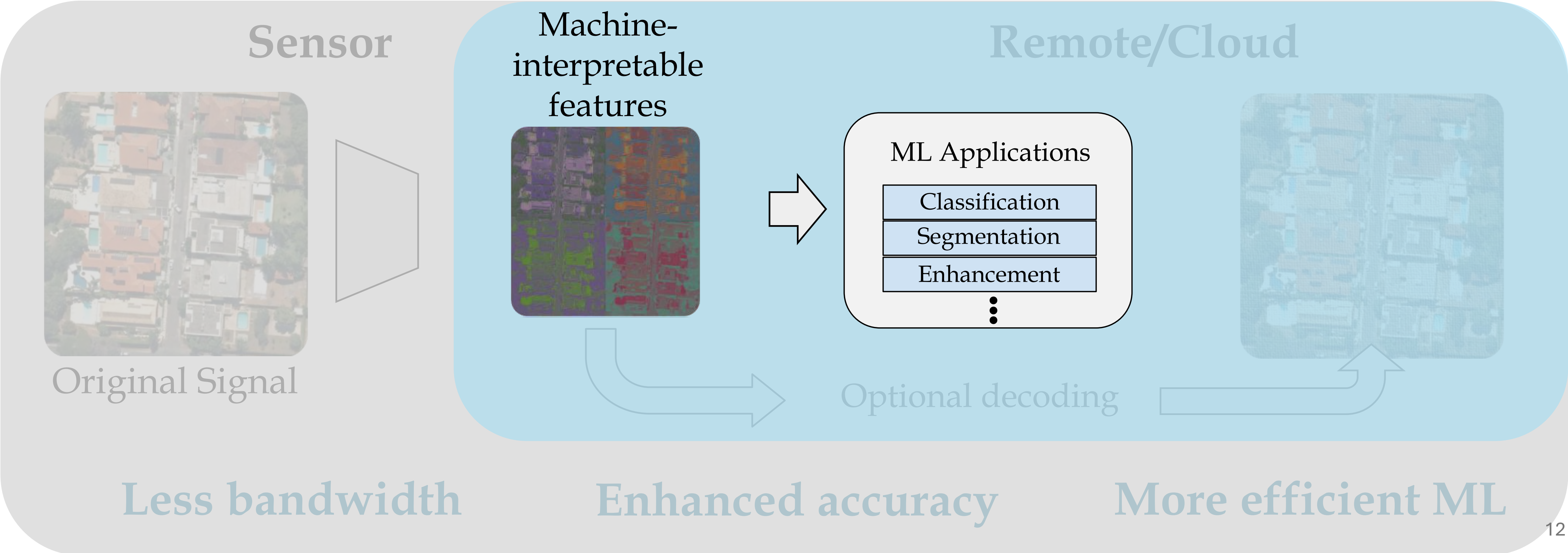
- Support many modalities
- Allow efficient encoding
- Preserve details
- **Achieve high compression rate**



Machine-oriented compression

What are ideal characteristics of the compression system?

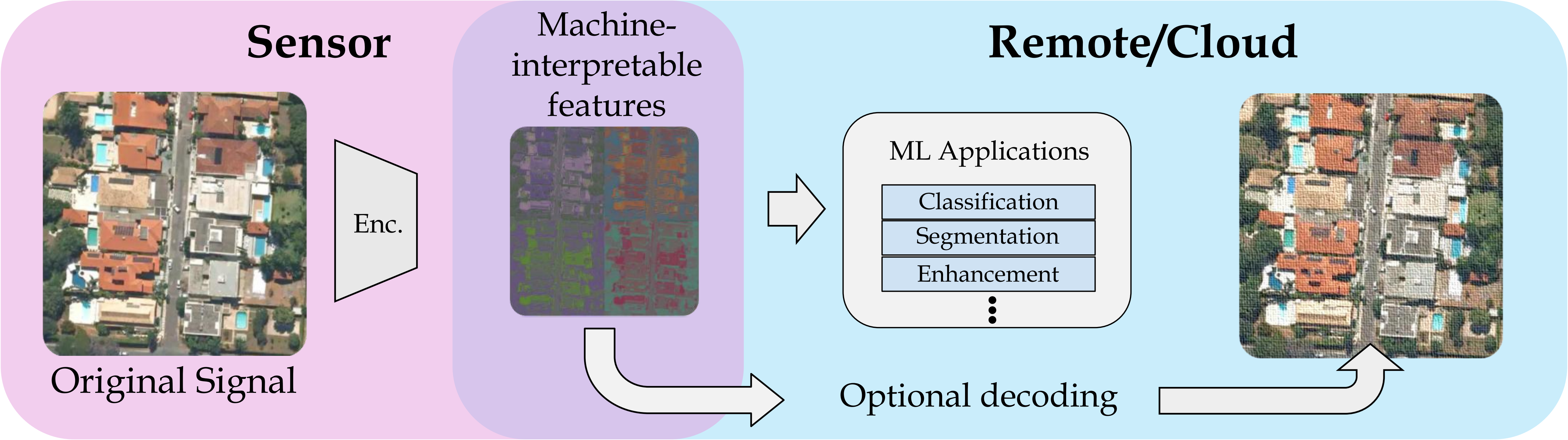
- Support many modalities
- Allow efficient encoding
- Preserve details
- Achieve high compression rate
- **Accelerate downstream ML models**



Machine-oriented compression

What are ideal characteristics of the compression system?

- Support many modalities
- Allow efficient encoding
- Preserve details
- Achieve high compression rate
- Accelerate downstream ML models



Less bandwidth **Enhanced accuracy** **More efficient ML**

Comparison of existing codec designs

Resample

	RR
Allow efficient encoding	✅

Comparison of existing codec designs

Resample

	RR
Allow efficient encoding	✅
Accelerate downstream ML	✅

Comparison of existing codec designs

Resample

	RR
Allow efficient encoding	✅
Accelerate downstream ML	✅
Achieve high compression rate	❌

Comparison of existing codec designs

Resample

	RR
Allow efficient encoding	✅
Accelerate downstream ML	✅
Achieve high compression rate	❌
Preserve details	❌

Comparison of existing codec designs

Resample

	RR
Allow efficient encoding	✅
Accelerate downstream ML	✅
Achieve high compression rate	❌
Preserve details	❌
Support many modalities	✅

Comparison of existing codec designs

Resample WEBP

	RR	LTC
Allow efficient encoding	✅	✅
Accelerate downstream ML	✅	❌
Achieve high compression rate	❌	✅
Preserve details	❌	✅
Support many modalities	✅	❌

Comparison of existing codec designs

Resample

WEBP

DGML (Cheng2020)

	RR	LTC	E2ELC
Allow efficient encoding	✅	✅	❌
Accelerate downstream ML	✅	❌	❌
Achieve high compression rate	❌	✅	✅
Preserve details	❌	✅	✅
Support many modalities	✅	❌	✅

Comparison of existing codec designs

Resample

WEBP

DGML (Cheng2020)

Stable Diff. VAE

	RR	LTC	E2ELC	GenAE
Allow efficient encoding	✅	✅	❌	❌
Accelerate downstream ML	✅	❌	❌	✅
Achieve high compression rate	❌	✅	✅	❌
Preserve details	❌	✅	✅	❌
Support many modalities	✅	❌	✅	❌

Comparison of existing codec designs

Resample

WEBP

DGML (Cheng2020)

Stable Diff. VAE

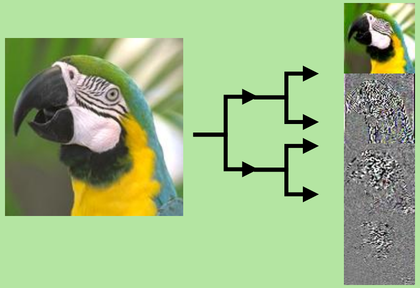
	RR	LTC	E2ELC	GenAE	Goal
Allow efficient encoding	✅	✅	❌	❌	✅
Accelerate downstream ML	✅	❌	❌	✅	✅
Achieve high compression rate	❌	✅	✅	❌	✅
Preserve details	❌	✅	✅	❌	✅
Support many modalities	✅	❌	✅	❌	✅

Proposed design

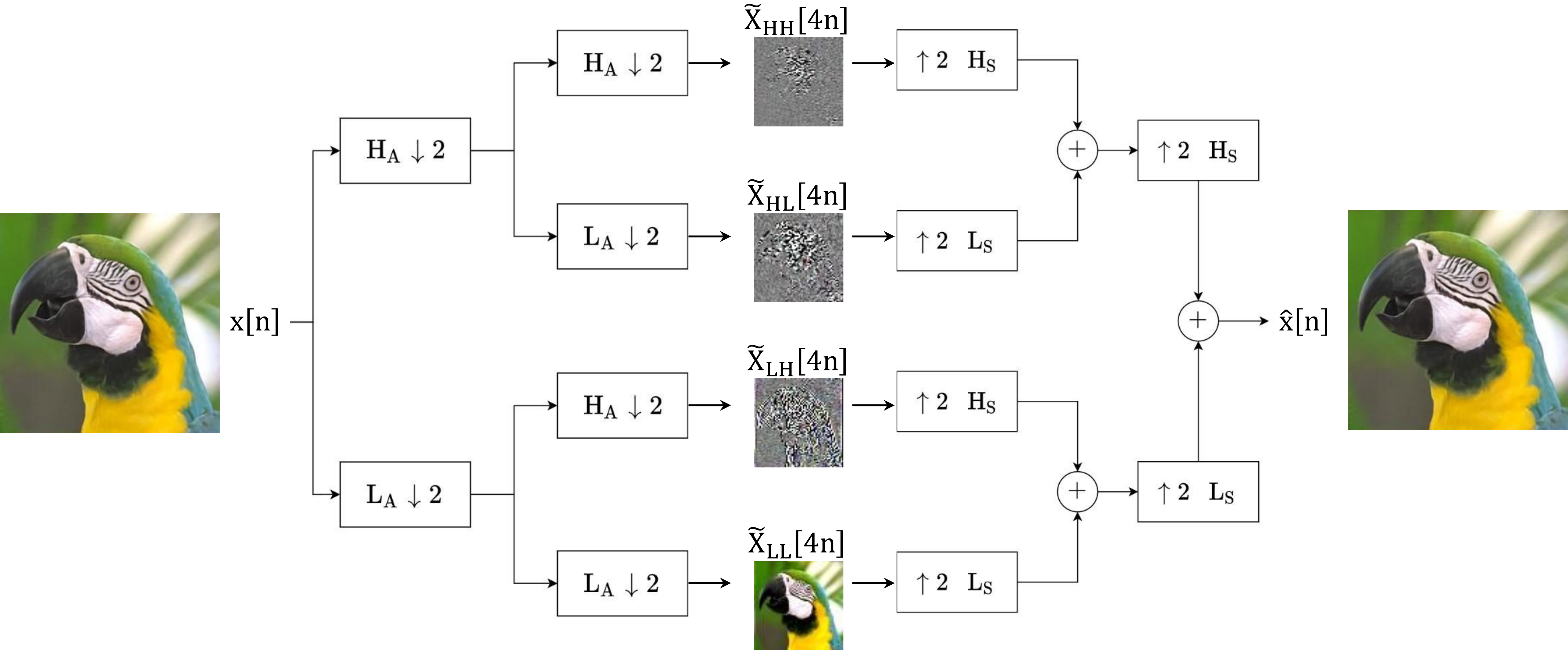
Encoding efficiency

Inspired by linear transform coding

Forgo expensive DNN-based analysis transform; leverage efficient, separable transform for energy compaction instead (wavelet packet decomposition)



Wavelet packet transform



WPT exchanges spatial resolution with channels

No information loss

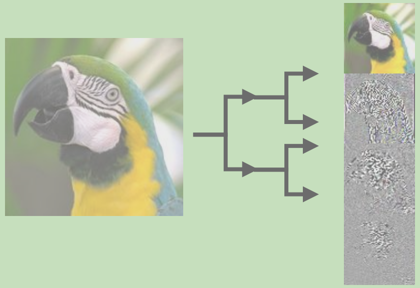
Energy compaction

Proposed design

Encoding efficiency

Inspired by linear transform coding

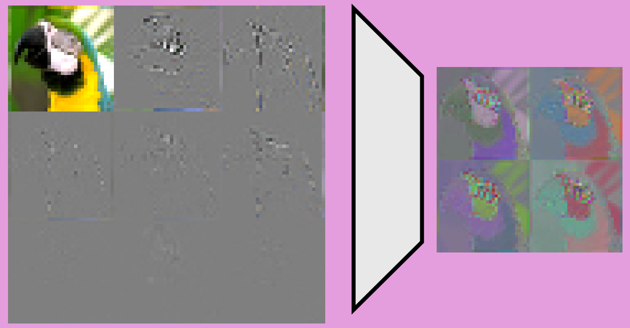
Forgo expensive DNN-based analysis transform; leverage efficient, separable transform for energy compaction instead (wavelet packet decomposition)



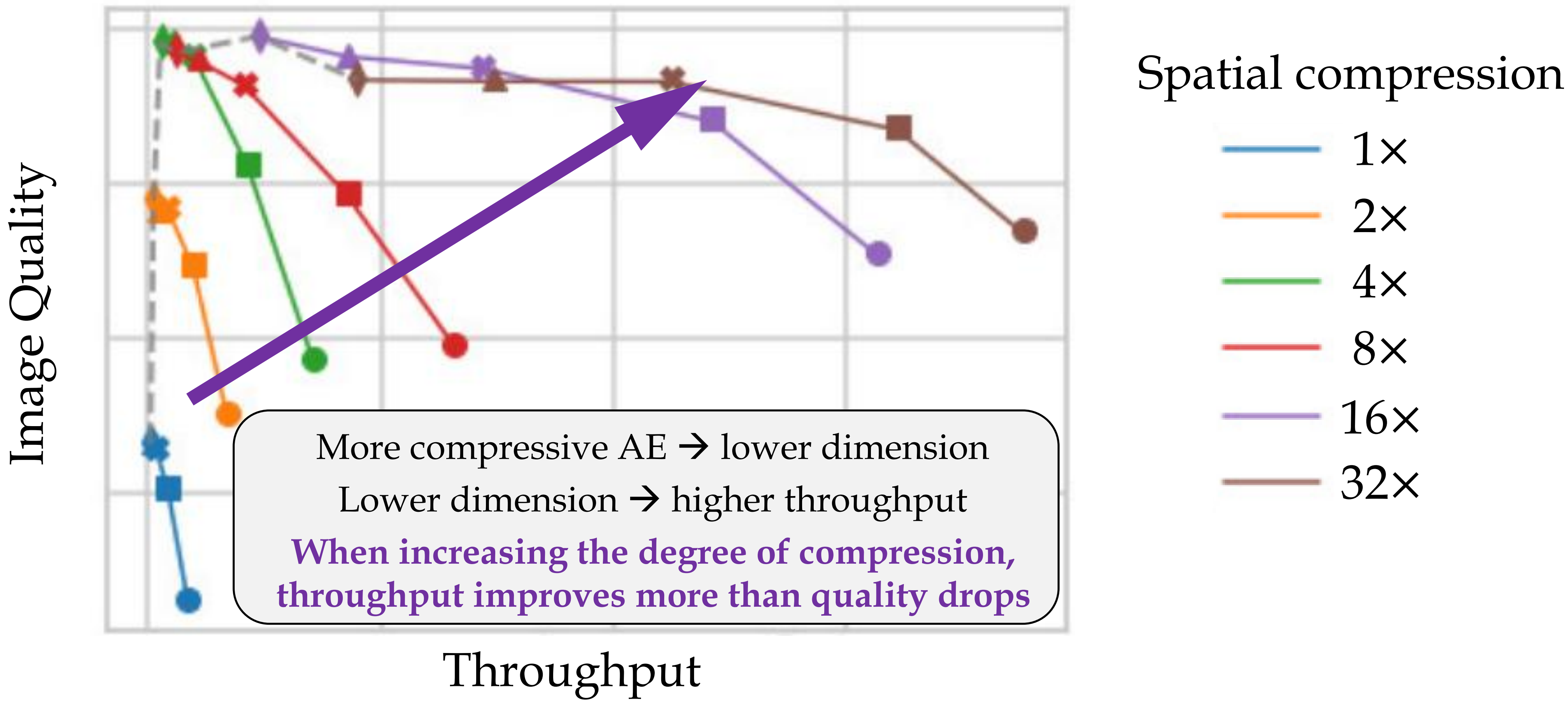
Dimension reduction

Inspired by generative autoencoders

Don’t rely exclusively on sparsity; use channel bottleneck to provide guaranteed, uniform dimensionality reduction to accelerate downstream models

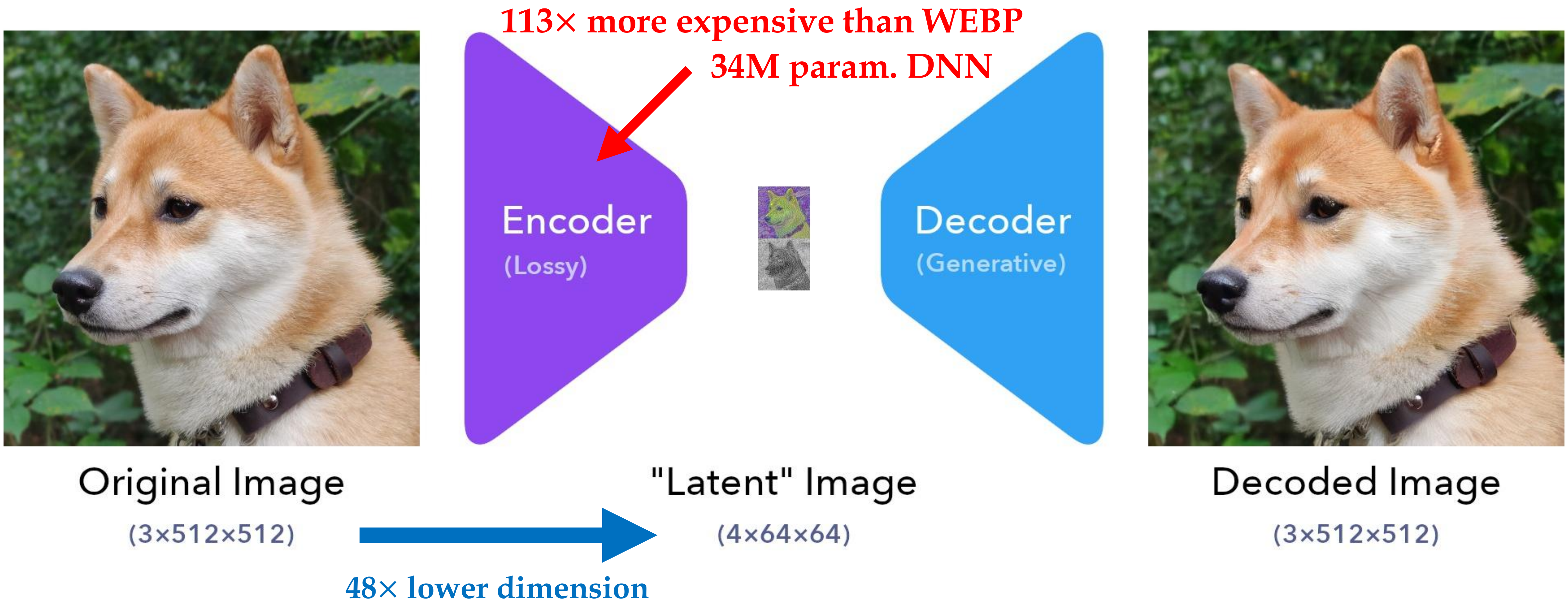


Autoencoder for dimension reduction



"High-Resolution Image Synthesis with Latent Diffusion Models"
(aka "Stable Diffusion") Rombach et al. 2021

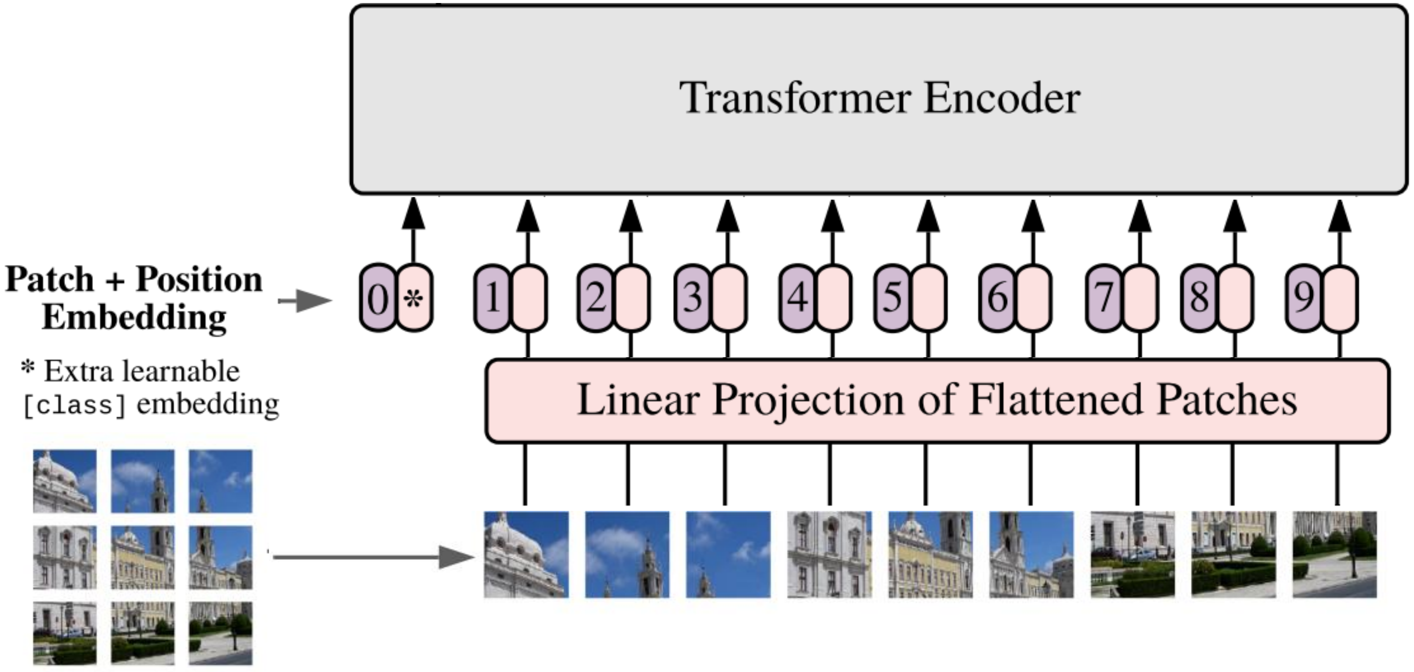
Autoencoder for dimension reduction



“High-Resolution Image Synthesis with Latent Diffusion Models”
(aka “Stable Diffusion”) Rombach et al. 2021

Does the encoder need to be so expensive?

Synthesizing details is hard
Discarding details is easy
→Use a simple encoder (e.g. linear projection)



"An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale" (aka "ViT") Beyer et al. 2021

ViT-B/16

Patch size	3×16×16
Sequence Len	196
Embedding Dim	768
Compression	1:1
Accuracy	86.1

ViT-B/32

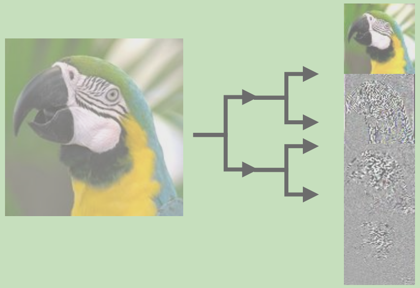
Patch size	3×32×32
Sequence Len.	49
Embedding Dim	768
Compression	4:1
Accuracy	83.3

Proposed design

Encoding efficiency

Inspired by linear transform coding

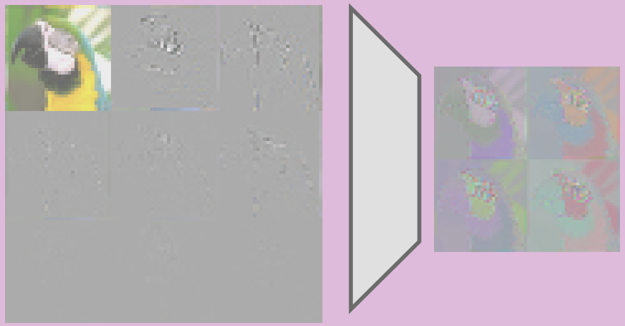
Forgo expensive DNN-based analysis transform; leverage efficient, separable transform for energy compaction instead (wavelet packet decomposition)



Dimension reduction

Inspired by generative AEs

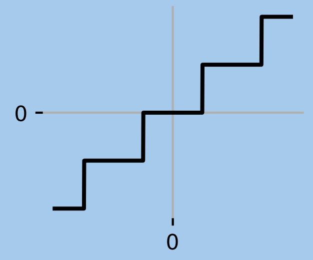
Don't rely exclusively on sparsity; use channel bottleneck to provide guaranteed, uniform dimensionality reduction to accelerate downstream models



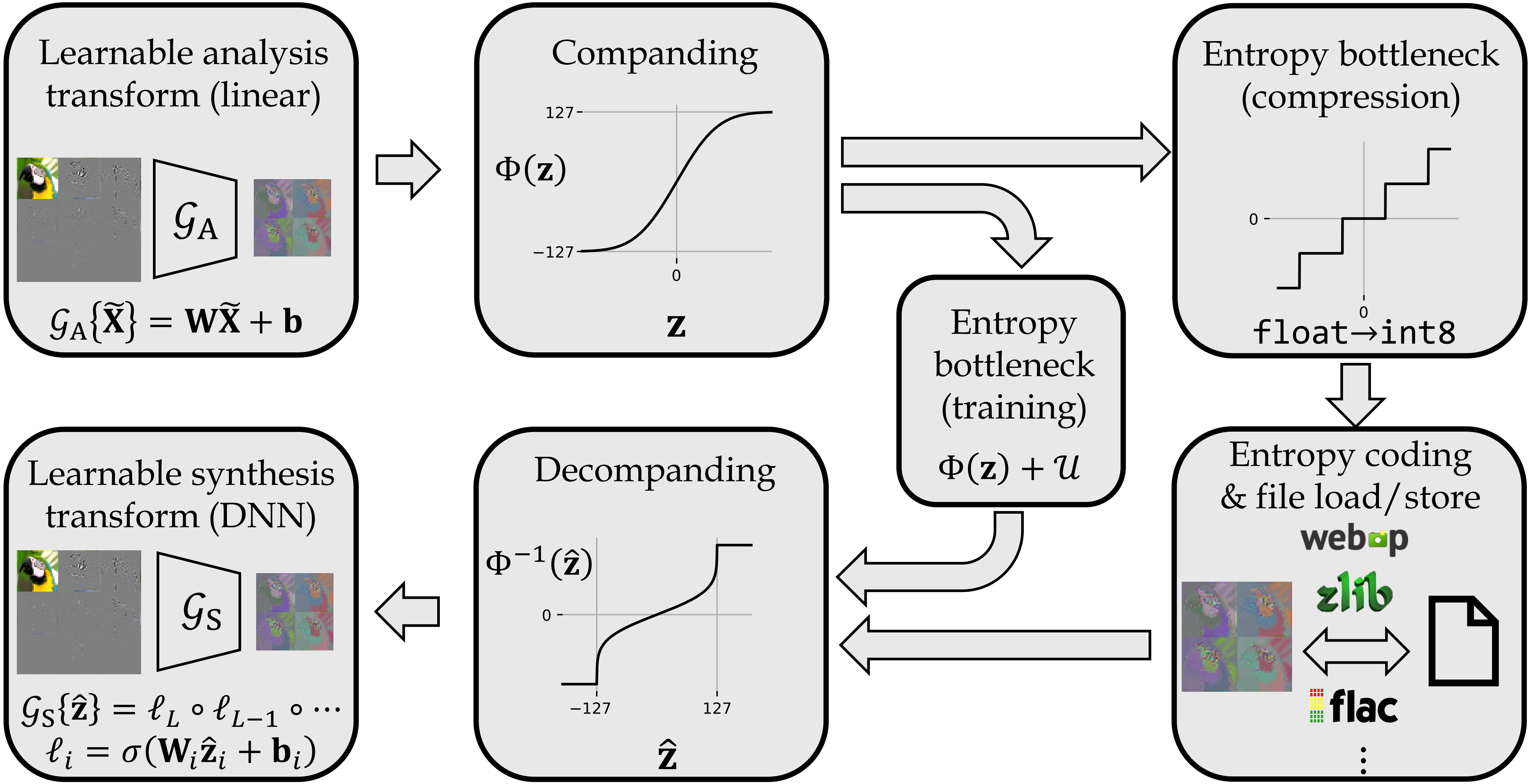
Compression ratio

Inspired by E2E learned compression

Guarantee resilience to quantization via additive noise during training. Leverage existing lossless codecs as a compression multiplier.



E2E learned compression: quantization and entropy coding

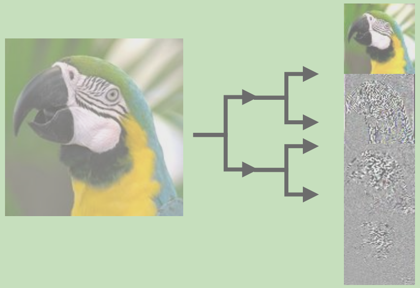


Proposed design

Encoding efficiency

Inspired by linear transform coding

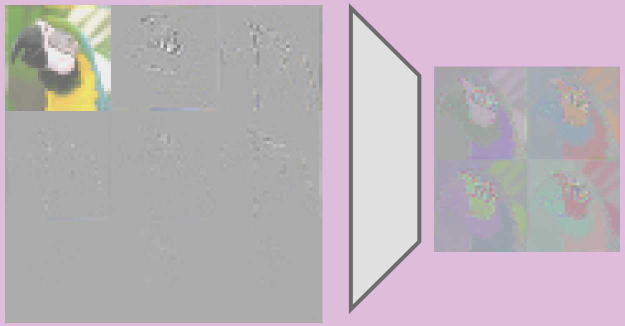
Forgo e
analysi
efficien
energy compaction instead
(wavelet packet decomposition)



Dimension reduction

Inspired by generative AEs

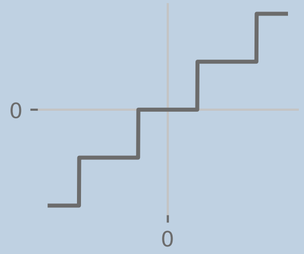
dimensionality reduction to
accelerate downstream models



Compression ratio

Inspired by E2E learned compression

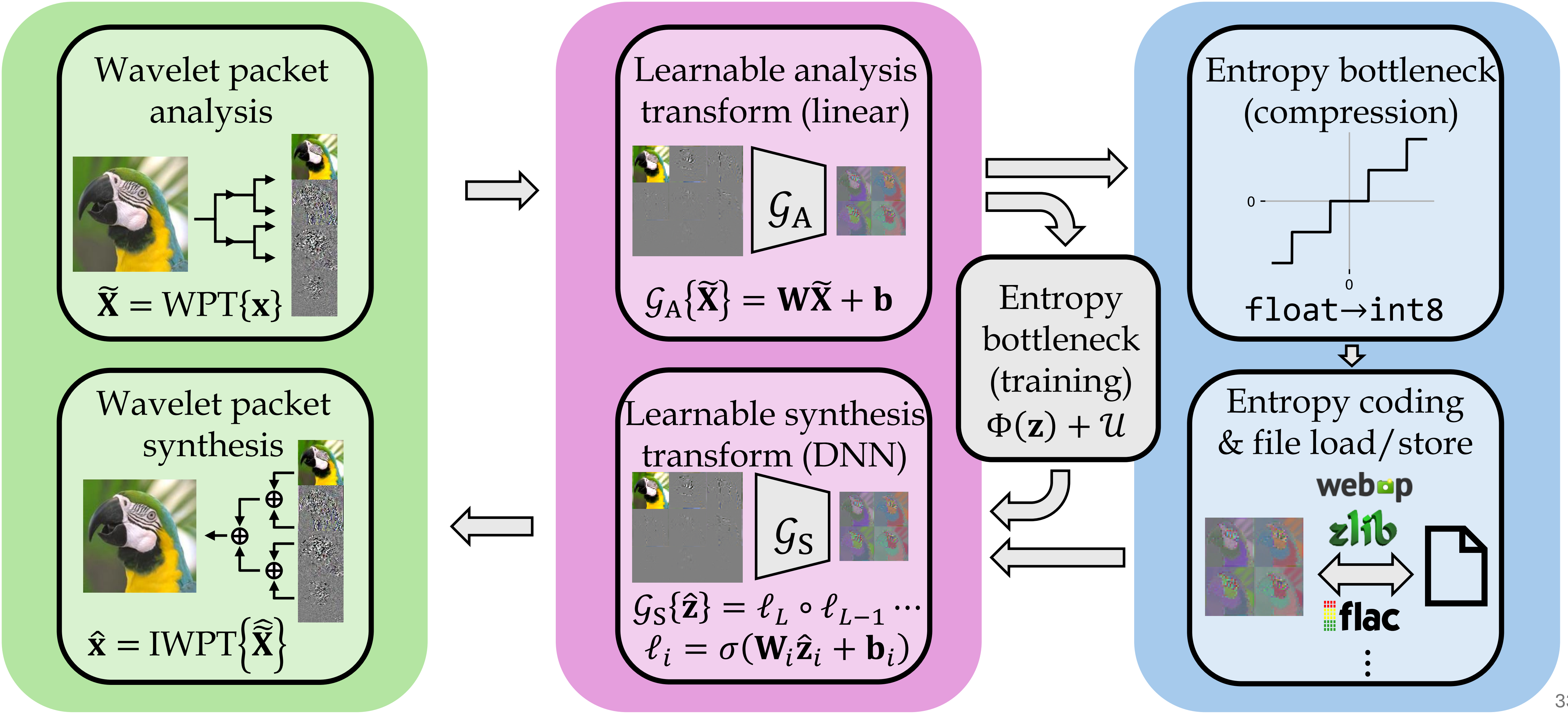
e noise
ge
existing lossless codecs as a
compression multiplier.



WaLLoC: Wavelet Learned Lossy Compression

WaLLoC workflow

Encoding efficiency | **Dimension reduction** | **Compression ratio**



How to avoid the pitfalls of generative autoencoders?

Resample | WEBP | DGML (Cheng2020) | Stable Diff. VAE

	RR	LTC	E2ELC	GenAE	Goal
Allow efficient encoding	✅	✅	❌	❌	✅
Accelerate downstream ML	✅	❌	❌	✅	✅
Achieve high compression rate	❌	✅	✅	❌	✅
Preserve details	❌	✅	✅	❌	✅
Support many modalities	✅	❌	✅	❌	✅

Loss function

$$\mathcal{L}(x,\hat{x}) = \cancel{\underbrace{\mathrm{MSE}(\mathrm{LPF}\{x\},\mathrm{LPF}\{\hat{x}\})}_{\substack{\text{Pooled MSE}\\ \text{(does not penalize high frequencies)}}}} + \cancel{\underbrace{\mathcal{L}_{\mathrm{LPIPS}}(x,\hat{x})}_{\substack{\text{Lerned perceptual}\\ \text{patch similarity}}}} + \cancel{\underbrace{\mathcal{L}_{\mathrm{GAN}}(x,\hat{x})}_{\substack{\text{Adversarial loss using}\\ \text{VGG16 discriminator}}}}$$

Only preserves low frequency details

Requires pre-trained models specific to RGB images

$$\mathcal{L}(x,\hat{x}) = \mathrm{MSE}(x,\hat{x})$$

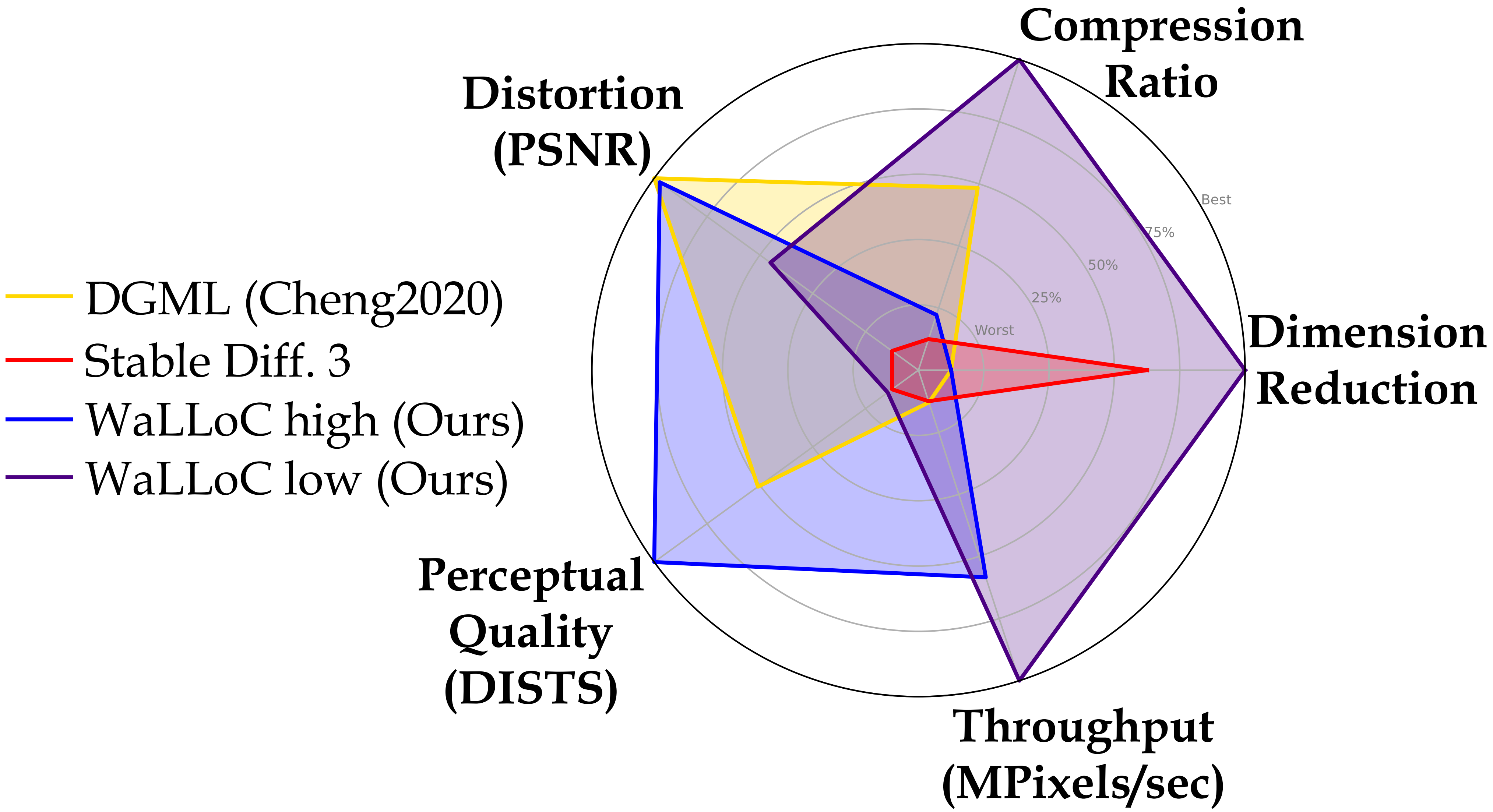
$$\hat{x} = \mathrm{decode}(\mathrm{encode}(x) + \mathcal{U})$$

Better preservation of high frequency details

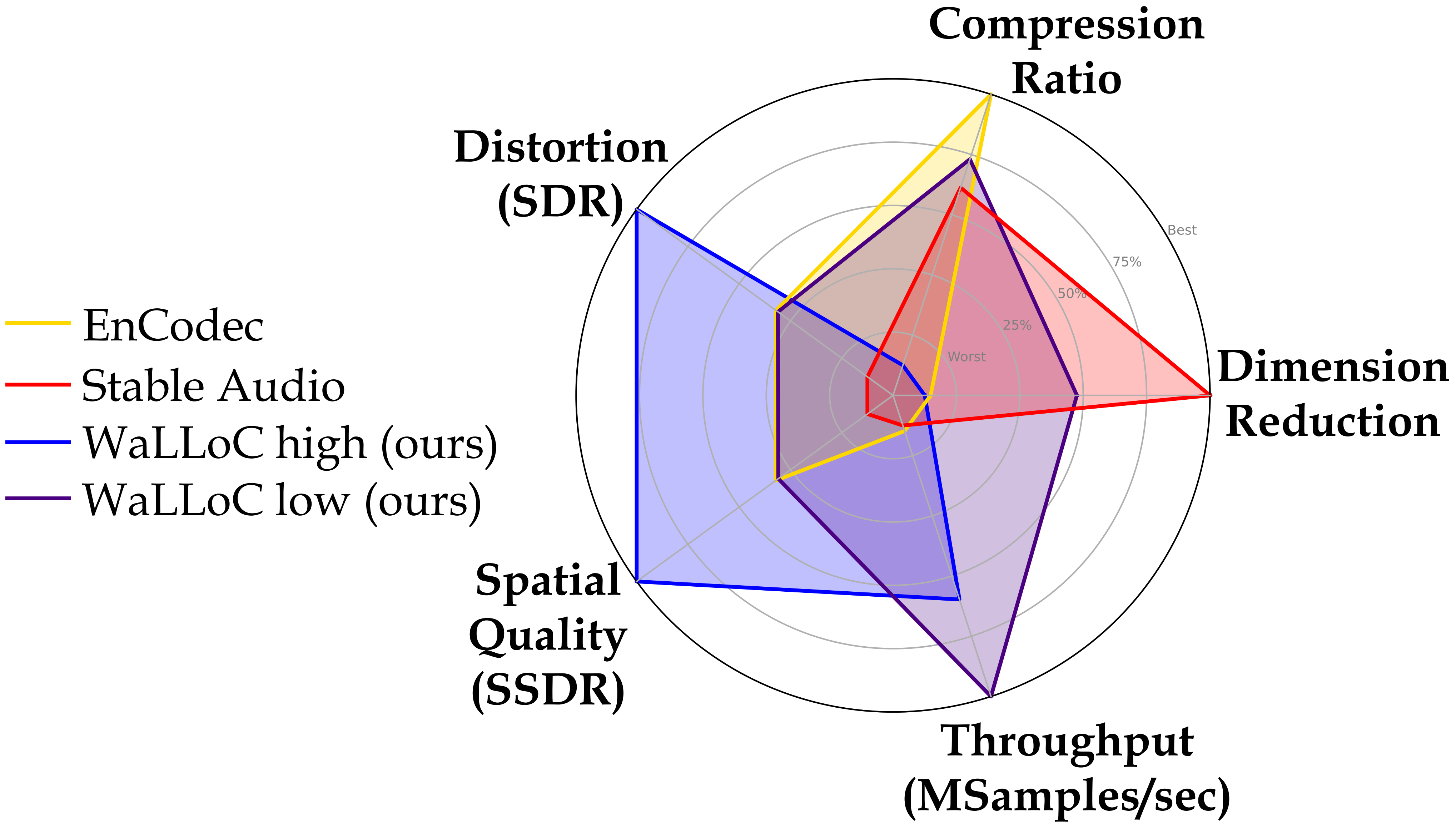
Supports a wide range of modalities

"Training VQGAN and VAE, with detailed explanation"
S. Ryu, 2024. github.com/cloneofsimo/vqgan-training

Comparison of autoencoder designs (RGB image)



Comparison of autoencoder designs (stereo audio)



How does it perform on downstream applications?

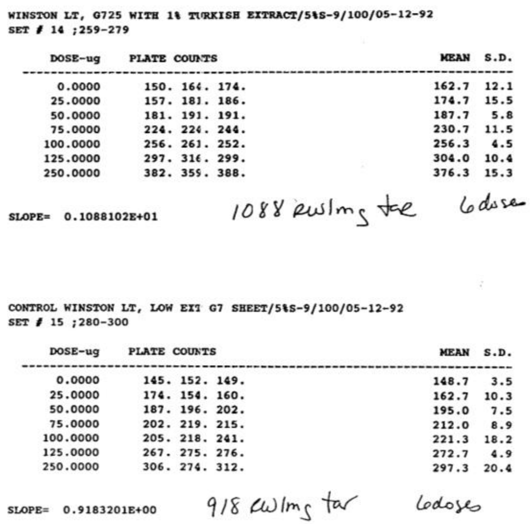
Image Classification

⇨ Cat

Document Understanding

Q: What is the date mentioned in the second table?

A: [“05-12-92”]



WINSTON LT, G725 WITH 1% TURKISH EXTRACT/5%S-9/100/05-12-92
SET # 14 ;259-279

DOSE-ug	PLATE COUNTS	MEAN	S.D.
0.0000	150. 164. 174.	162.7	12.1
25.0000	157. 181. 186.	174.7	15.5
50.0000	181. 191. 191.	187.7	5.8
75.0000	224. 224. 244.	230.7	11.5
100.0000	256. 261. 252.	256.3	4.5
125.0000	297. 316. 299.	304.0	10.4
250.0000	382. 359. 388.	376.3	15.3

SLOPE= 0.1088102E+01 1088 revs/mg tar 6dose

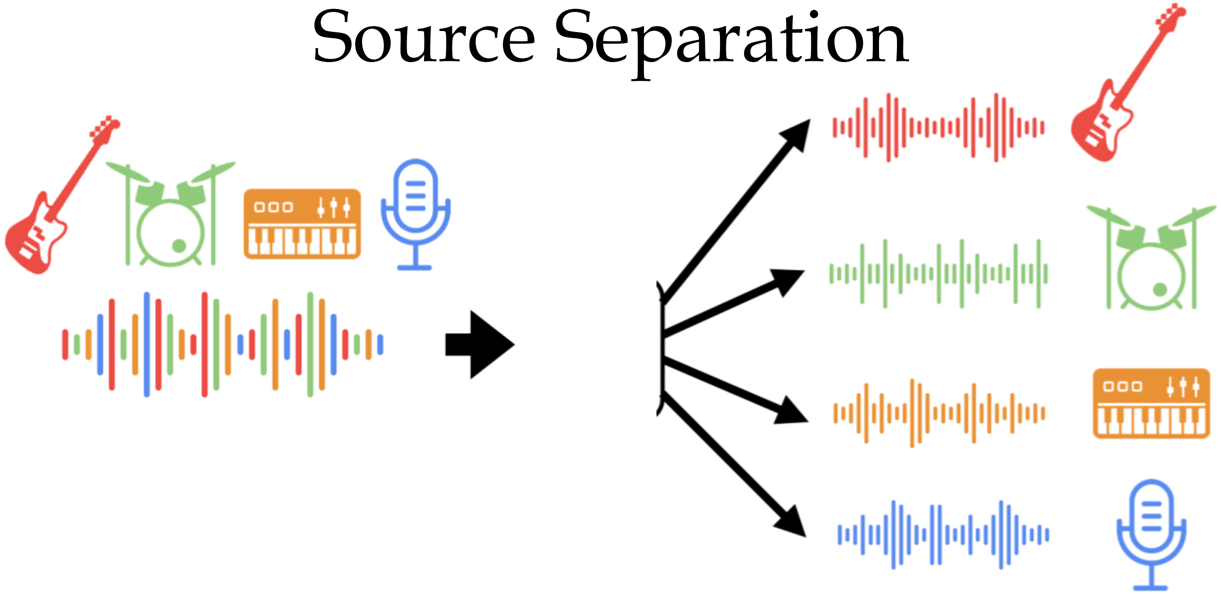
CONTROL WINSTON LT, LOW EXT G7 SHEET/5%S-9/100/05-12-92
SET # 15 ;280-300

DOSE-ug	PLATE COUNTS	MEAN	S.D.
0.0000	145. 152. 149.	148.7	3.5
25.0000	174. 154. 160.	162.7	10.3
50.0000	187. 196. 202.	195.0	7.5
75.0000	202. 219. 215.	212.0	8.9
100.0000	205. 218. 241.	221.3	18.2
125.0000	267. 275. 276.	272.7	4.9
250.0000	306. 274. 312.	297.3	20.4

SLOPE= 0.9183201E+00 918 revs/mg tar 6doses

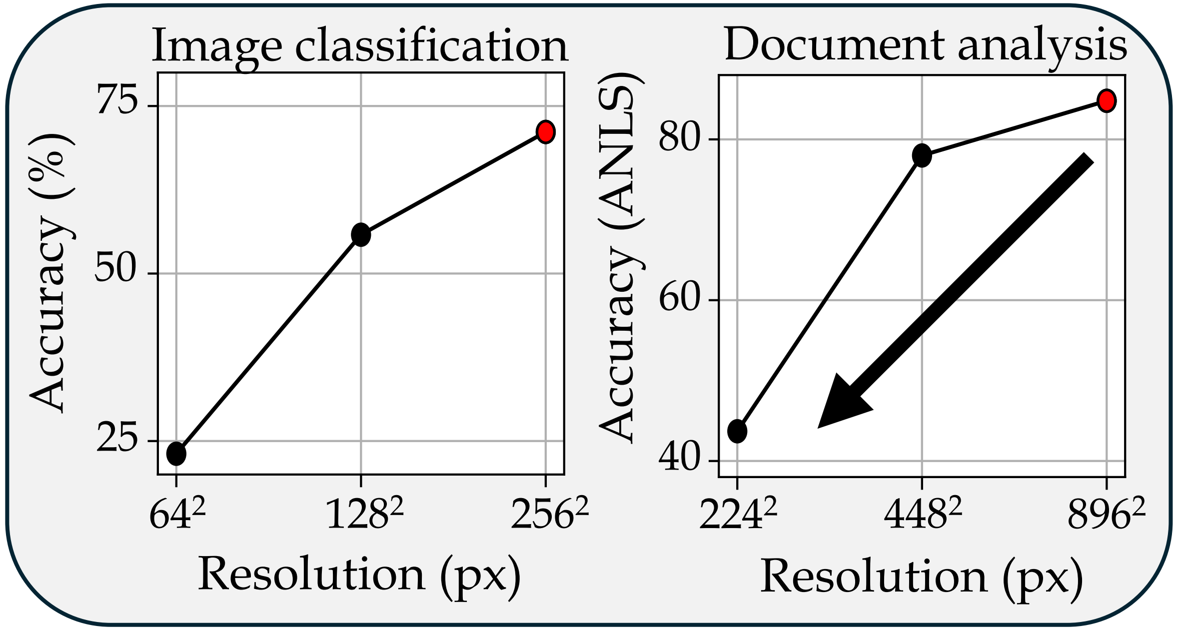
Colorization

Source Separation



Comparison vs. resolution reduction

Reduced resolution & reduced computation



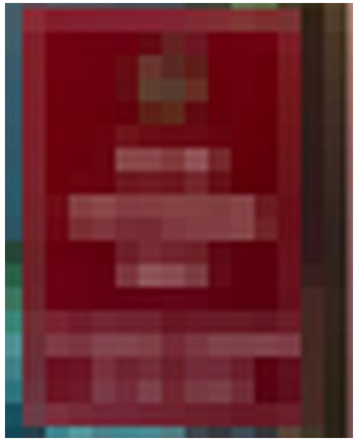
4× lower latency

21GB→8GB GPU Mem

85% → 44% Accuracy

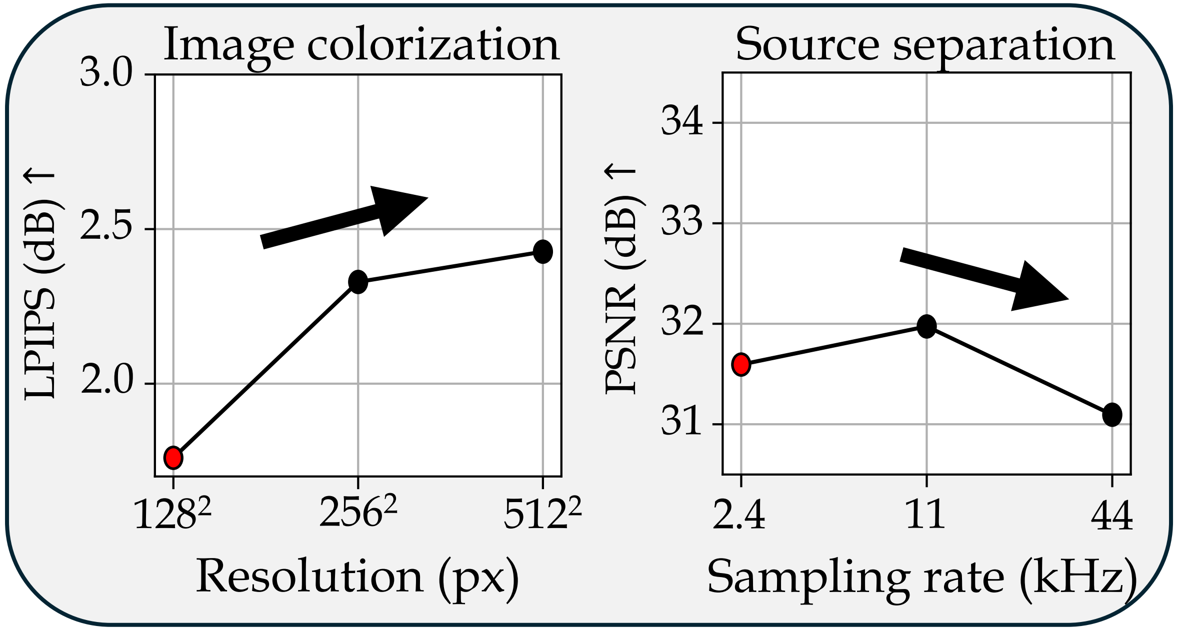
Baseline ●

Resample —●—



Comparison vs. resolution reduction

Increased resolution & fixed computation



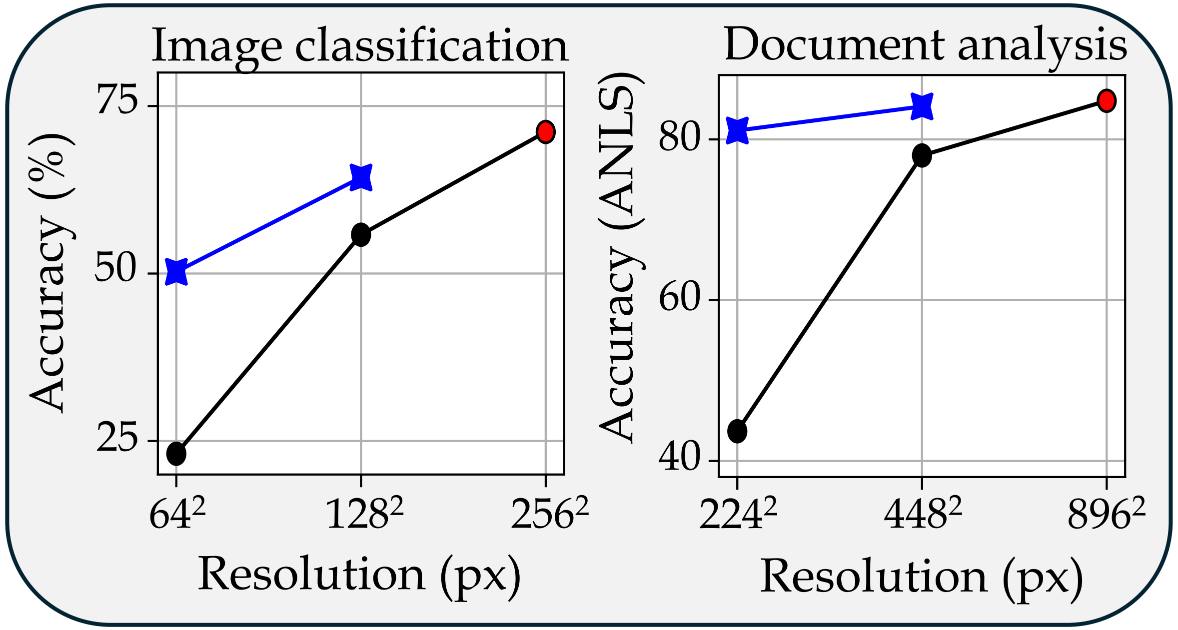
Larger patches

Diminishing return of larger patches / filters

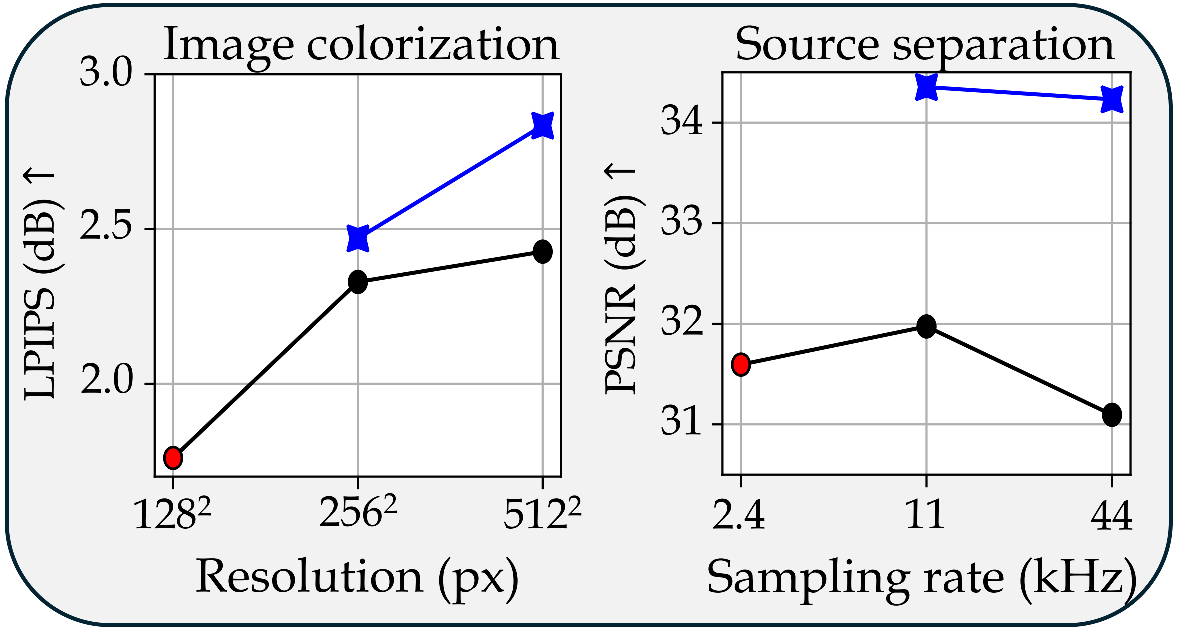
Baseline

Comparison vs. resolution reduction

Reduced resolution & reduced computation

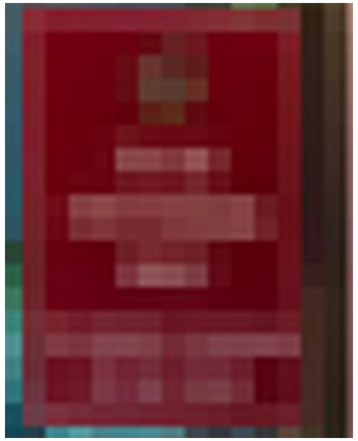


Increased resolution & fixed computation



Baseline

Pixels

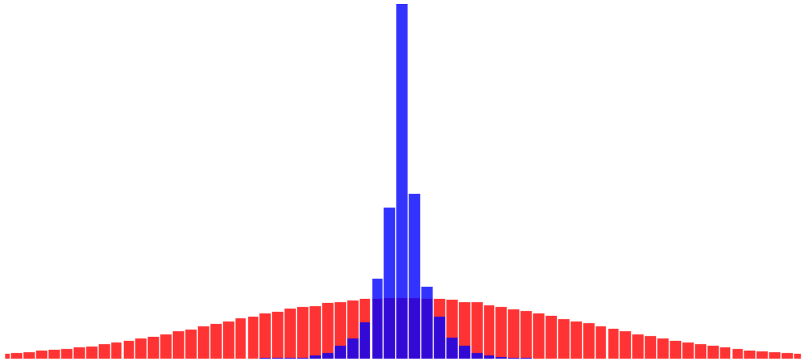


Ours

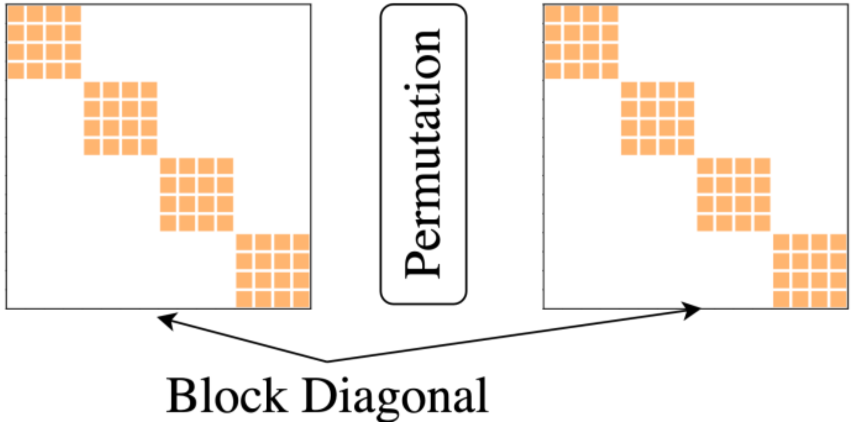
Extensions

Rate penalty

$$\text{Loss} = \text{MSE}(\mathbf{x}, \hat{\mathbf{x}}) + \log(\sigma)$$



Deep, nonlinear, but subquadratic encoder



Kodak	WEBP (Q=1)	Ours
CR ↑, DR ↑	145:1, 1×	153:1, 64×
PSNR ↑, SSIM ↑, DISTS ↑	27.18, 0.826, 7.50	27.18, 0.862, 9.01
CPU Encode throughput ↑	26.3 MP/sec	11.95 MP/sec

Try it for yourself!

Installation → `pip install walloc`

Audio→ [Pre-trained codec]

Images→ [Pre-trained codec]

Training (1D) → [Tutorial]

Training (2D) → [Tutorial]

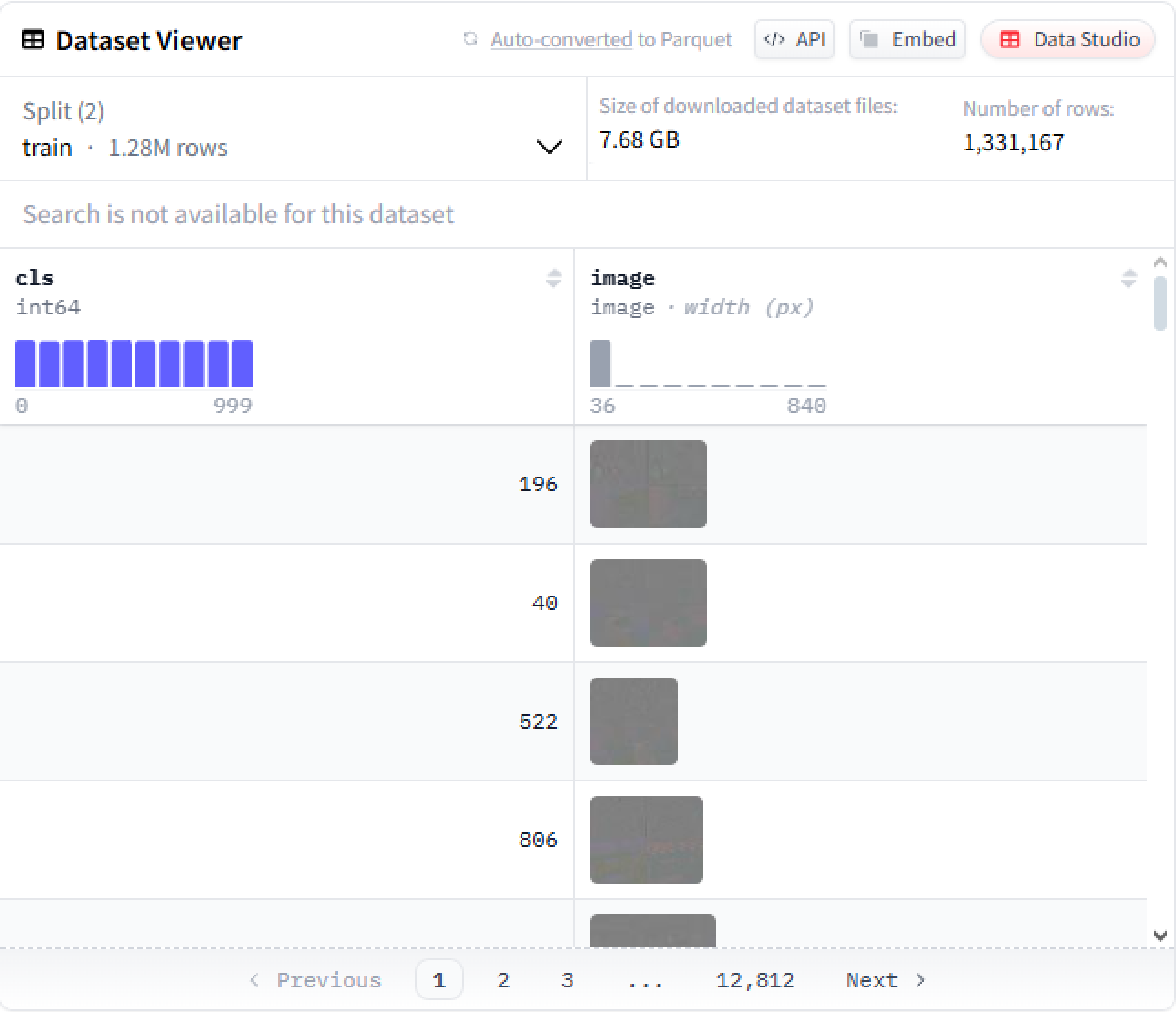
More details available:

https://ut-sysml.org/walloc/

Contact: danjacobellis@utexas.edu

Backup

Compatibility with ML frameworks



hf.co/datasets/danjacobellis/inet1k_compressed

WEBP

64× fewer "pixels"

9.3 kB
22.9 dB PSNR

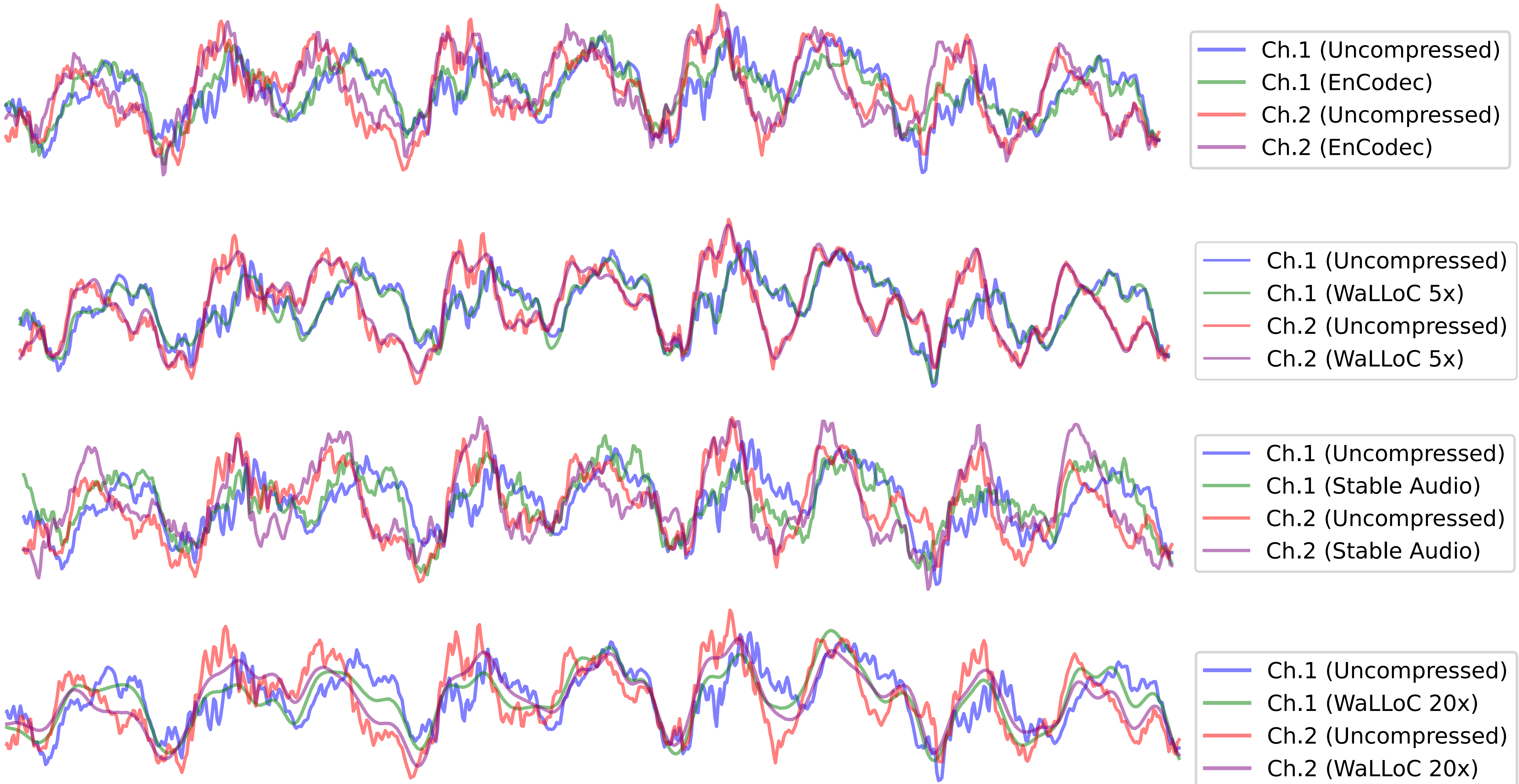
WaLLoC

64x96

Use like a normal image dataset

```
ds = load_dataset("danjacobellis/inet1k_compressed")
compressed_batch = ds.select(range(256))
decoded_batch = []
for img in compressed_batch:
    decoded_batch.append(pil_to_tensor(img))
```

Stereo Audio



Impulse response