Semantic Preserving Image Compression
ICIP 2020

Session: COM-01

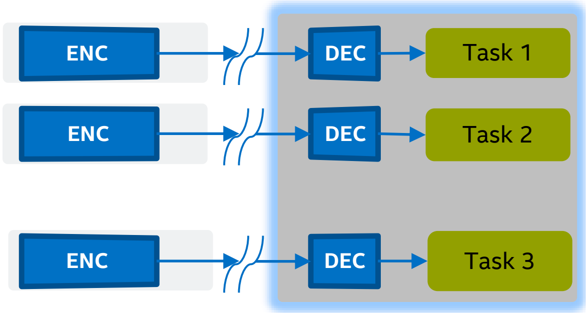
Presenter: Nilesh Ahuja

Authors: Neel Patwa (Intel[1]), Nilesh Ahuja (Intel), Omesh Tickoo (Intel), Srinivasa Somayazulu (Intel), Srenivas Varadarajan (Samsung[2]), Shashidhar Koolagudi (NIT Karnataka)

[1]Presently at Qualcomm
[2]Was at Intel Labs during this research

Intel Labs

Motivation: Enabling Distributed Analytics in Visual IoT space



Current analytics pipeline in Visual IoT

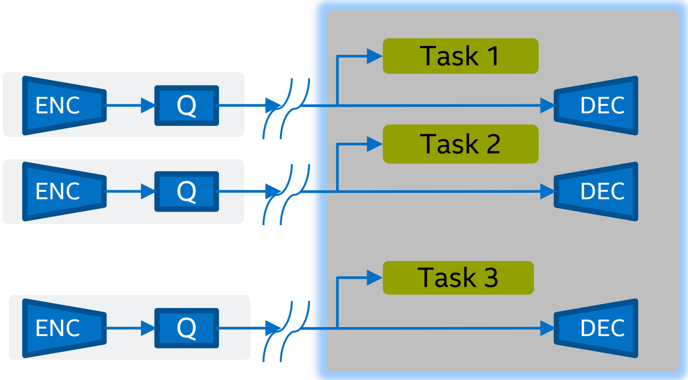
- In Visual Internet of Things (IoT) space, large amounts of visual data captured by low-power mobile/client devices needs to be transferred to the cloud for processing and analysis.
- Standard lossy compression techniques optimize perceptual quality rather than performance on visual analytic tasks

Problem Statement

Compression of visual content to maximize performance on a visual analytic task (e.g. classification, detection, etc.)

intel

Proposed Approach: ML-Based Compression

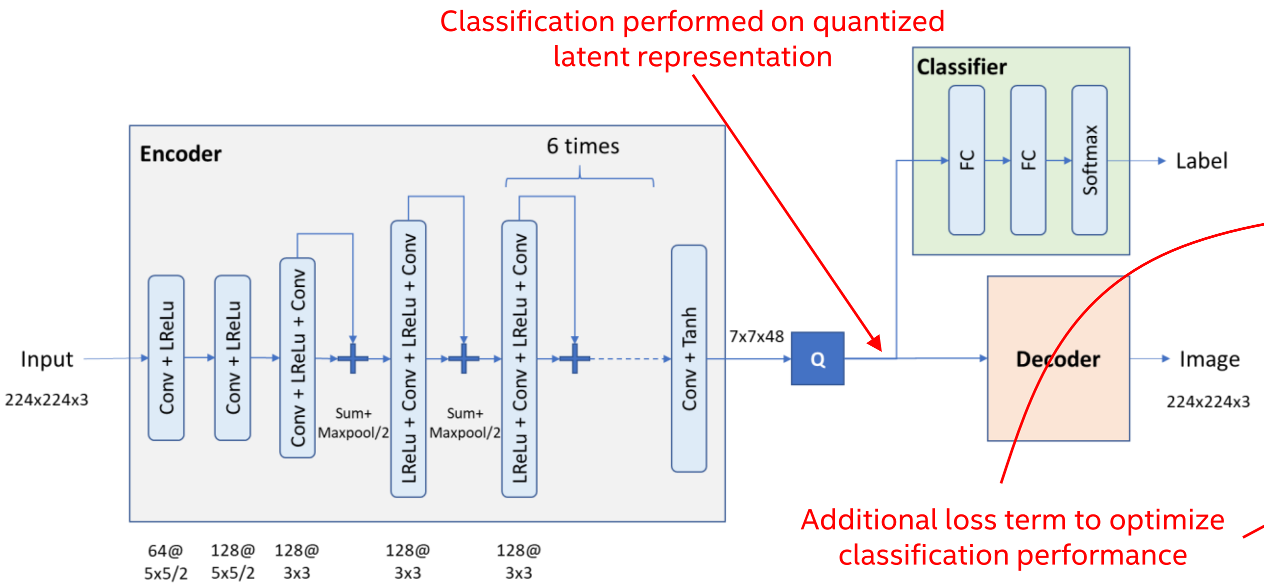


Proposed analytics pipeline

Machine-learning Based Compression

- Learn optimal representations for a given task.
- Task can be performed directly on learned representation without decoding
- Use of task-specific distortion measures allows rate-distortion optimization for that task

Semantic Preserving Image Compression



Convolutional Autoencoder architecture

- $N@M \times M/S$ = N channels, kernel-size $M \times M$, stride S.
- LReLu = leaky ReLu.
- Q = Quantization + lossless (Huffman) encoding

Training

- Trained E2E with multi-task training loss

$$L = L_d + \alpha L_r + \beta L_c$$

- Distortion loss L_d = MSE (mean-squared error)
- Rate loss $L_r = -\log q([f(x) \cdot \lambda])$, where $\lambda = 1/QF$ (quantization factor)
- Classification loss L_c= Cross-entropy loss
- Non-differentiability of quantization function Q(x):
  - Perform true quantization in forward pass.
  - Approximate Q by identity during back-prop.

Experimental Setup

- Measure classification accuracy as a function of compression level indicated by bits-per-pixel (BPP)
- Dataset: ImageNet
- Baselines:
  - Classification accuracy on JPEG-compressed images on three different architectures: ResNet, VGG19, and our architecture.
  - DeepSIC[1]

Table of Quantization Factors

SPIC QF	JPEG QF	Avg BPP
128	46	0.143
48	30	0.112
16	25	0.100

To enable a fair comparison, the quantization levels for both JPEG and SPIC are adjusted so that average BPP across test-set is same for both.

[1]Sihui Luo, Yezhou Yang, Yanling Yin, Chengchao Shen, Ya Zhao, and Mingli Song, "DeepSIC: Deep semantic image compression," in *International Conference on Neural Information Processing*. Springer, 2018, pp. 96–106

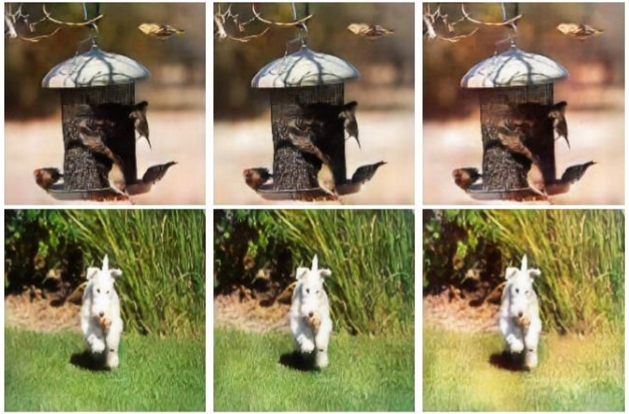
RESULTS

Reconstruction Performance: ImageNet

Original

0.143 BPP 0.112 BPP 0.100 BPP

JPEG compressed



0.143 BPP 0.112 BPP 0.100 BPP

SPIC compressed (Our approach)

Reconstruction Performance: Kodak CD

Original

BPP 0.136
SSIM=0.888

Original

BPP=0.143
SSIM=0.819

Quality of reconstructed images is good despite the images being from a completely different dataset!

SPIC vs JPEG

Classification accuracy at various compression levels

BPP	With SPIC-Q	With JPEG Compression		
		SPIC-U	Resnet50	VGG19
N/A	73.31	73.31	**74.9**	71.3
0.143	**72.51**	66.31	65.27	63.16
0.112	**69.86**	63.01	62.59	60.88
0.100	**63.27**	60.93	61.13	59.42

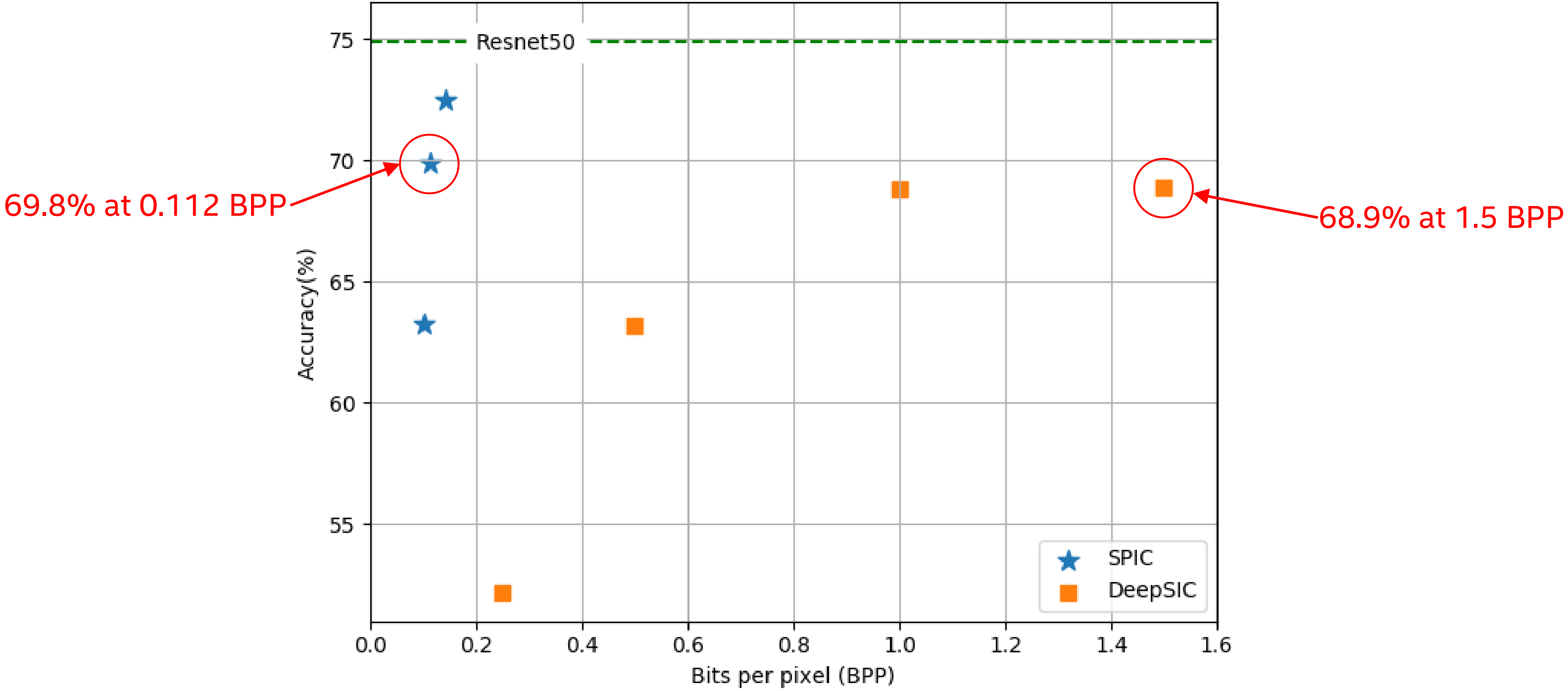
Quality of image reconstruction

BPP	SPIC		JPEG	
	SSIM	PSNR	SSIM	PSNR
0.143	0.847	22.28	**0.921**	**23.74**
0.112	0.815	21.54	**0.901**	**22.58**
0.100	0.741		**0.891**	

- SPIC outperforms JPEG across all compression levels on all tested architectures.
- Visual quality of the images reconstructed by our method is lower than that of JPEG

Supports our original hypothesis that perceptually-significant visual features might not be the most suitable for classification tasks

SPIC vs DeepSIC



Future Work

- Explore various autoencoder architecture (e.g. RNNs).
- Adaptive, task-aware latent-space decomposition
- Use of more sophisticated lossless coding schemes (arithmetic coding) to reduce the bit rate
- Extending the concept to other tasks like such as object detection and tracking.

Thank You !

Intel Labs

Intel Confidential — Internal Use Only