Tampere University

Comprehensive Complexity Assessment of Emerging Learned Image Compression on CPU and GPU

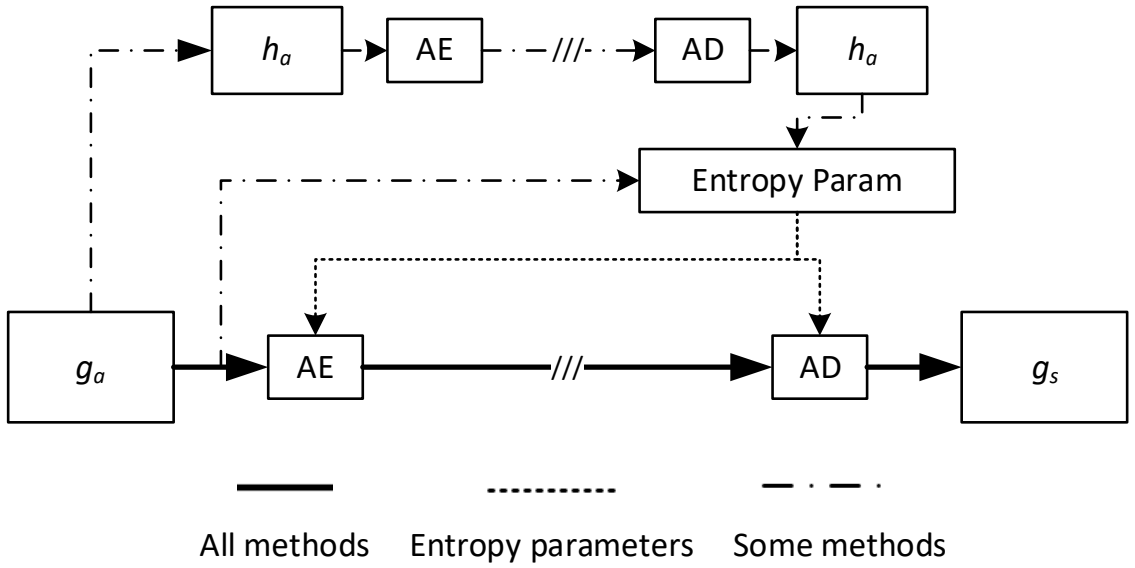
Farhad Pakdaman, and Moncef Gabbouj

IEEE ICASSP 2023

This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No [101022466]

Learned Image/Video Compression

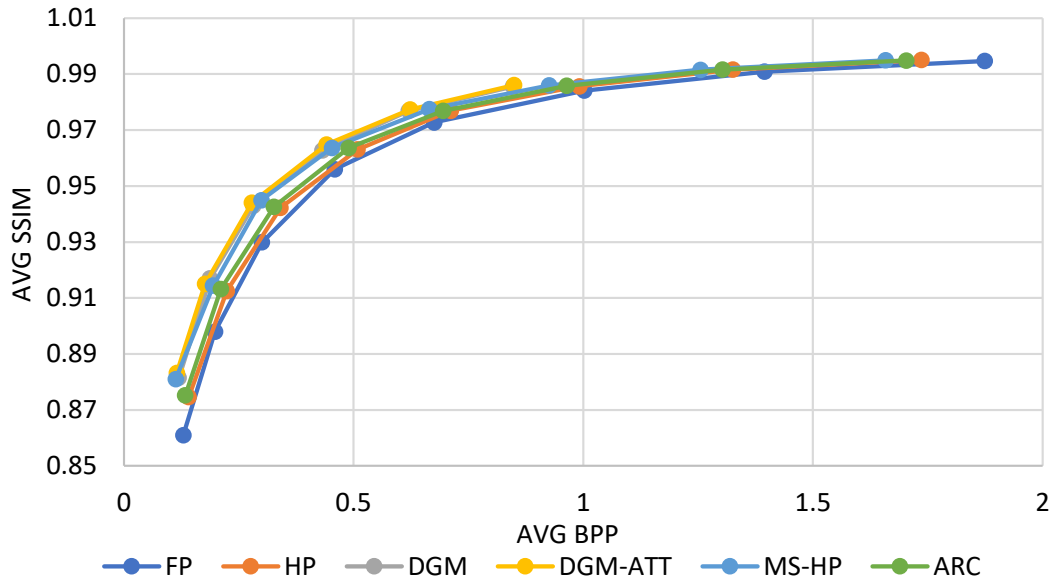
- Compression using a deep-neural network
- End-to-end optimized/trained
- Still a young field, but great potentials
  - Already surpassed tradition compression
  - Targeted for both human and machine consumption
  - Pursued in upcoming standards JPEG-AI, MPEG VCM, MPEG NNVC



Assessed Methods

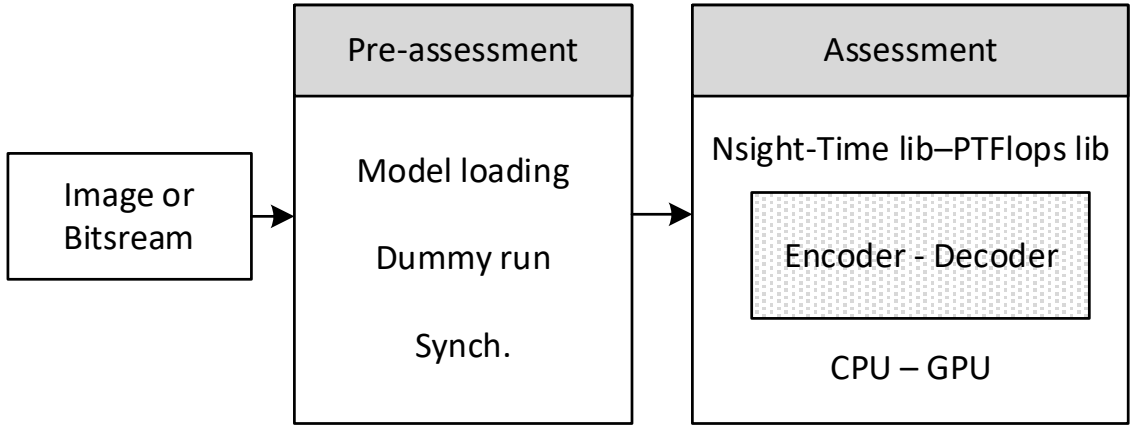
- Six methods – basis for many more methods
- Factorized prior (**FP**)
- Scale hyperprior (**HP**)
- Mean and scale hyperprior (**MS-HP**)
- Autoregressive context model (**ARC**)
- Discretized Gaussian Mixture Likelihood (**DGM**)
- DGM with attention mechanism (**DGM-ATT**)

Fig 10. Average rate-distortion performance of LC methods



Assessment Methodology

- Tested on CPU and GPU
- Warm-up stage: loads the codec, dummy run, and CUDA-synch
- Measurements by Nvidia Nsight System and Time library, and PTFlops library
- Methods from CompressAI library, each with 6 or 8 quality factors (Q)
- Kodak dataset



Overall Encoding Complexity

- Up to 12x difference in CPU and 50x in GPU time for different methods
- Complexity increases for higher Q (better qualities)
- Different GPU speedup for methods (max 8.1x for low-dependency, 2.5x high-dependency methods)
- 4.6x speedup for Enc, and 2.2x for Dec
- Up to 1025 kMAC/Pel for DGM-ATT

Fig 3. Total Enc times (s) on CPU. ARC, DGM, DGM-ATT on left, and FP, HP, and MS-HP on right axis

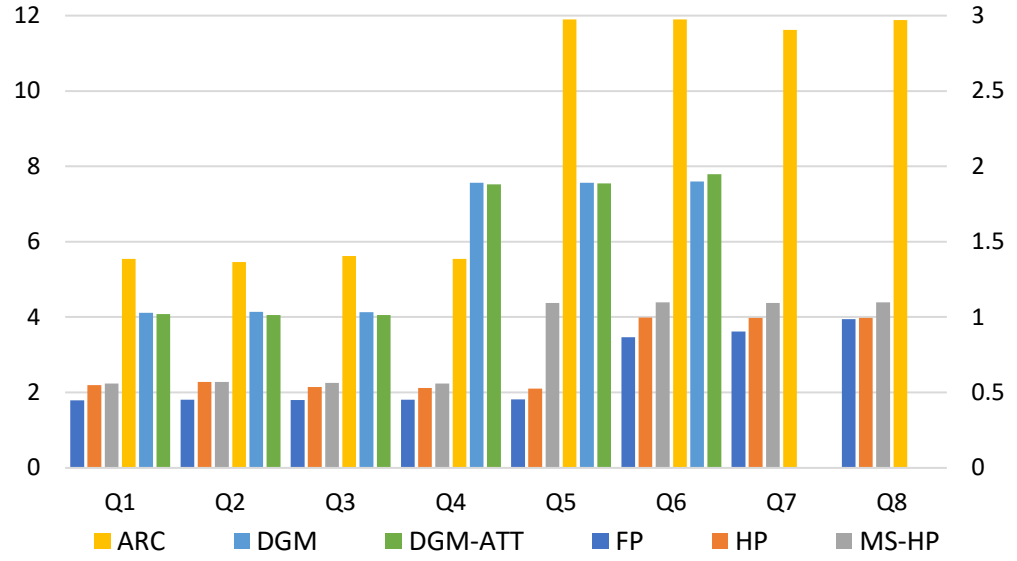
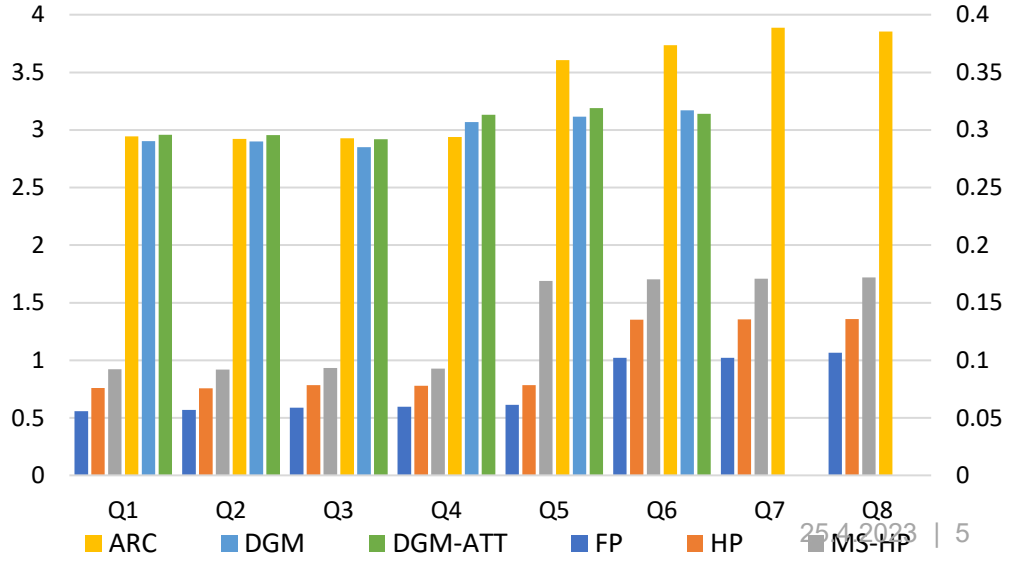


Fig 4. Total Enc times (s) GPU



Overall Decoding Complexity

- Similar Enc and Dec complexities

Fig 5. Total Dec times (s) CPU

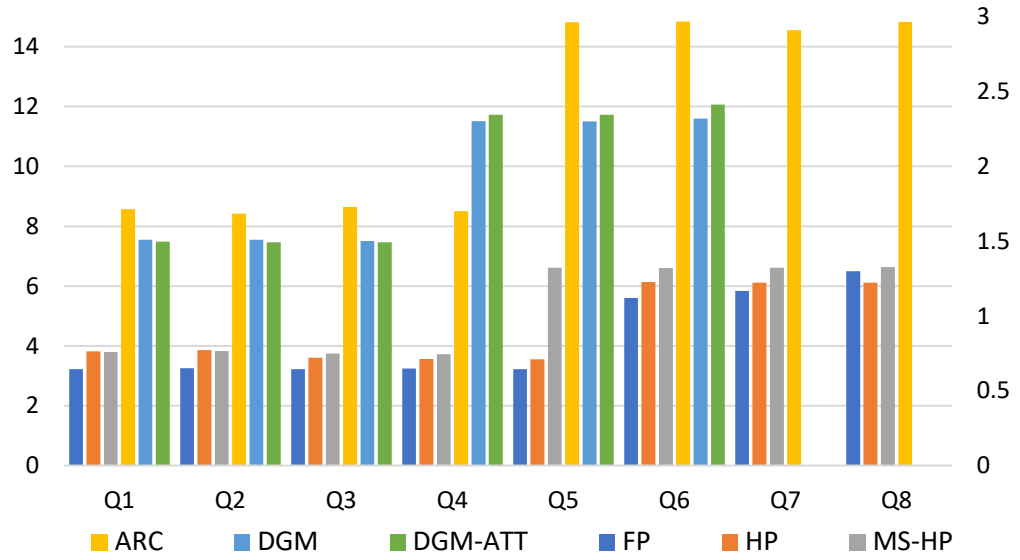
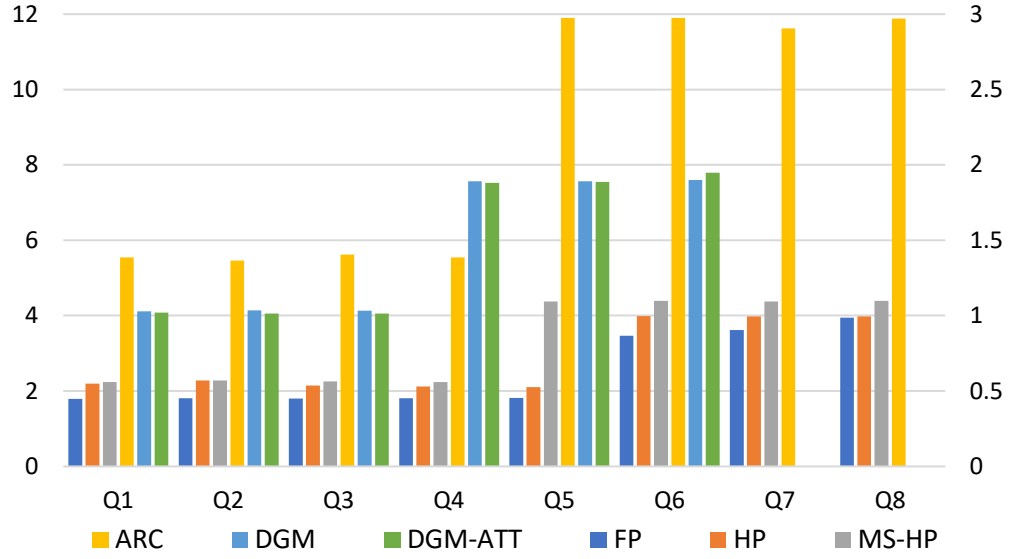


Fig 6. Total Dec times (s) GPU



Comparing with HEVC/VVC

- 4x up to 100x more complex than HEVC
  - Both Enc and Dec
- Much less content dependent
- No fast decision available

Table I. Average HEVC and VVC times (s) on CPU

	Enc	Dec
AVG HM (HEVC)	1.53 (std 0.23)	0.06 (std 0.006)
AVG VTM (VVC)	57.4 (std 17.03)	0.09 (std 0.01)

Coding Modules

- Analysis and synthesis transforms (ga and gs) take the major complexity on CPU, but minor on GPU
- AE and AD complex on both CPU and GPU
  - More contribution on GPU, due to data dependencies
- ReLU, Conc, and element-wise multiplication dominate Enc. Conv, GeMM, and element-wise multiplicaton for Dec.

Fig 7. Module time shares (%) for both CPU and GPU, Enc and Dec

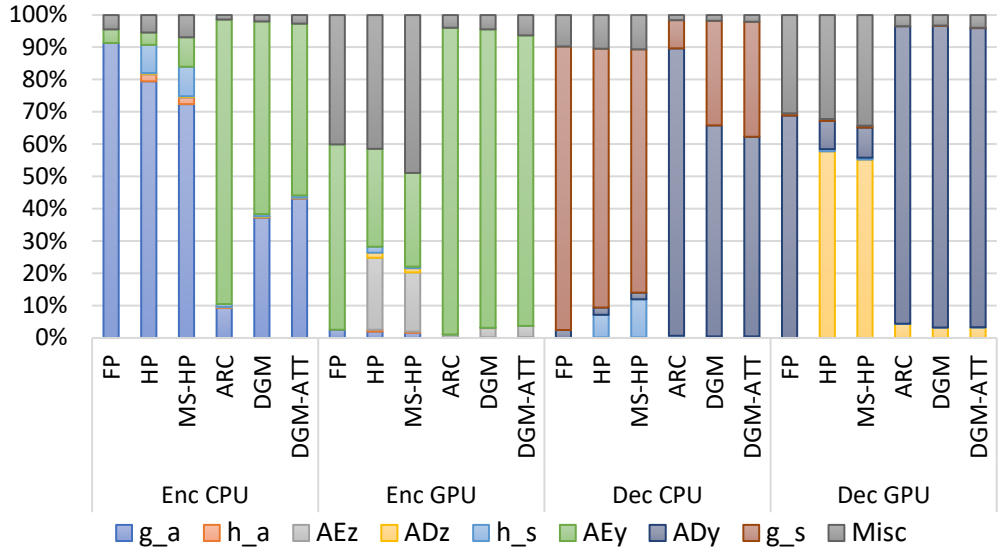
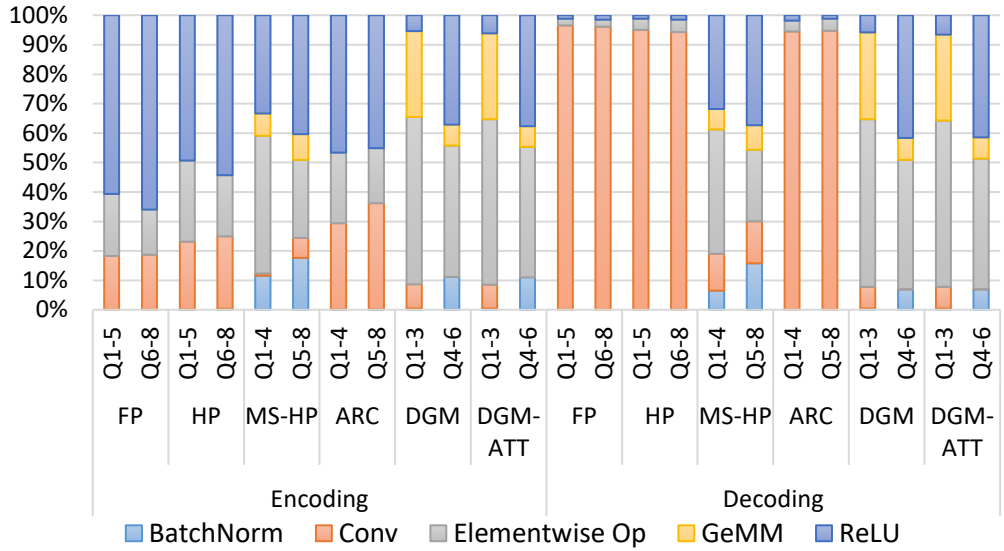


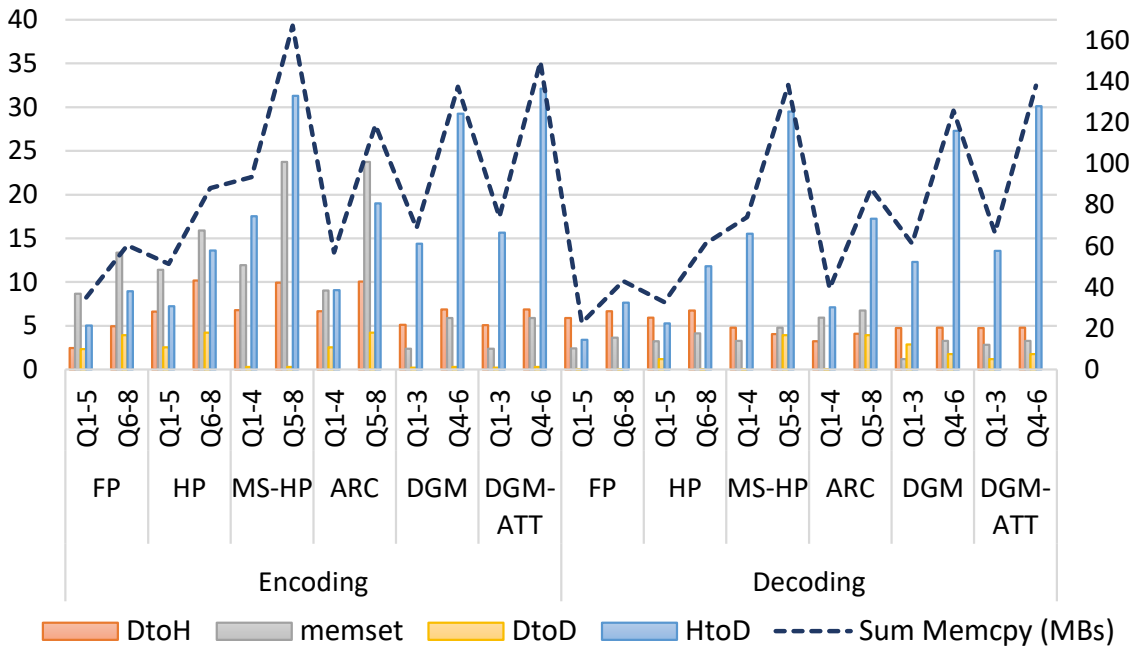
Fig 8. CUDA kernel shares (%) for Enc and Dec



Memory Footprint

- Higher Q correspond to higher memory requirements (between 1.7 to 2.2x)
- Autoregressive method and attention-based models are the most demanding
- Dec takes 0.65x to 0.92x of Enc

Fig 9. Memory requirements (MBs) for Enc and Dec. Sum memory and HtoD correspond to the right axis



Remarks

- The overall complexity of LC methods is much higher than traditional (HEVC and VVC). Unlike the traditional, decoding complexity of LC methods is close to their encoding complexity.
- Methods with a more complex context modeling have a significantly higher complexity at both encoding and decoding.
- Unlike the traditional, complexity of LC methods is almost content independent.
- LC methods have different parallelizability on GPU. Dec acceleration is lower than enc for all methods. Methods with high dependency in context modeling gain a limited speedup.
- Analysis transform and entropy coding are the most demanding encoding operations on CPU and GPU, respectively.
- Synthesis transform and entropy decoding are the most demanding decoding operations on CPU and GPU, respectively.
- On GPU, ReLU and convolution are the most used kernels for simpler methods, and elementwise operations and GeMM are for the more complex ones.
- Memory transfer from host to device is the largest memory usage, which almost doubles for the higher end of quality levels.

Thank You!

- Hope to meet some of you at the conference!
- More details and data will be available on GitHub:

https://github.com/farhad02/LC_Assessment