Neural Network Compression (NNC, ISO/IEC 15938-17)

Werner Bailer

AI and Multimedia Workshop, 2022-01-18

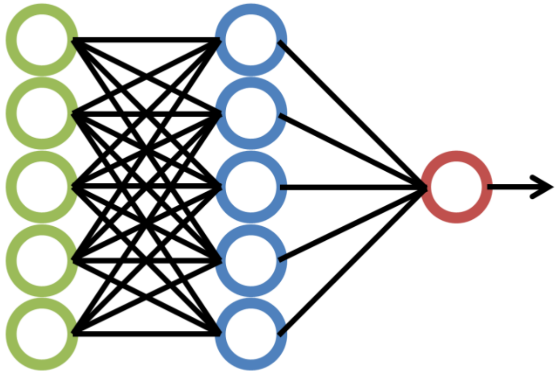
JOANNEUM RESEARCH DIGITAL

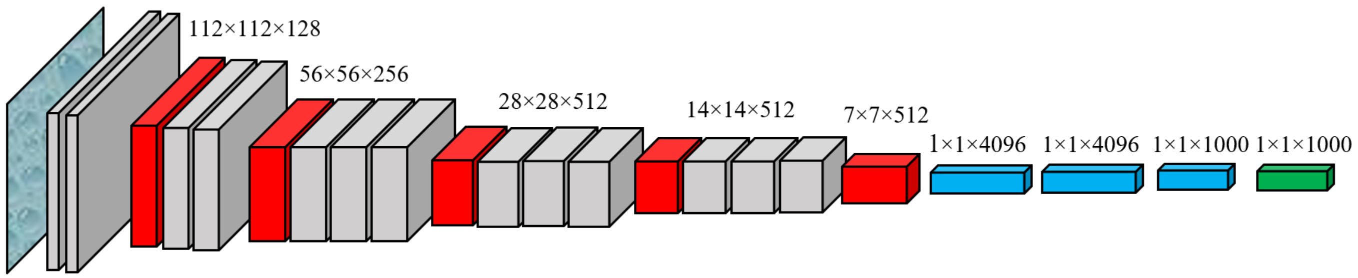
Outline

- Context & state of the art
- Standard: need and design considerations
- Coding tools
- Performance
- Ongoing work and conclusion

Context

- Artificial neural networks are widely used, e.g. in multimedia
  - visual and audio content recognition and classification
  - speech and natural language processing
- Deep learning makes use of very large networks
  - many layers with nodes and connections
  - parameters/weights attached to each of them (e.g. convolution operations)



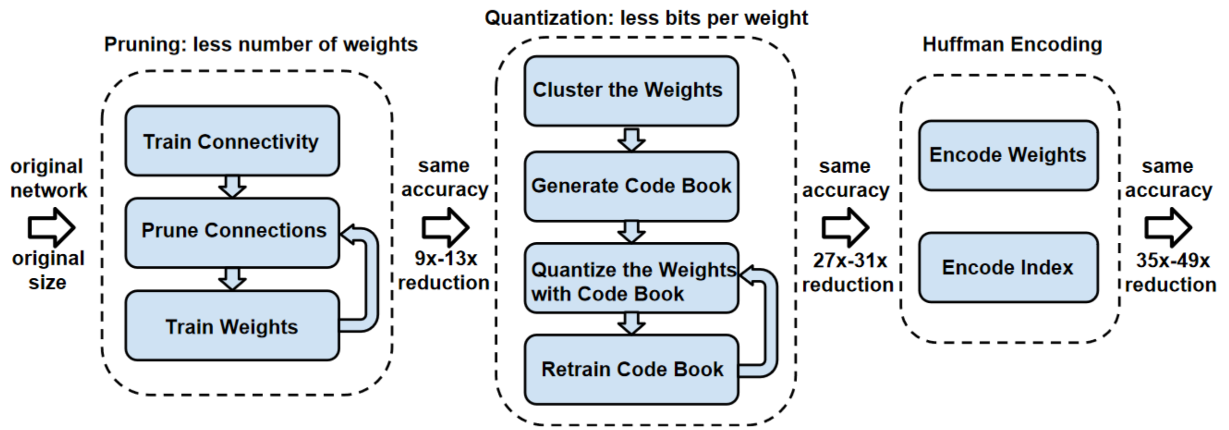


Context

- Training
  - learn parameters from data
  - typically once, on powerful infrastructure
  - updates or adaptations in target environment may be necessary
- Inference
  - use the trained network for prediction
  - needs network with all its parameters, i.e., large amount of data to be transmitted and processed
    → ***focus: small size to be transmitted***
  - often used on resource constrained devices (mobile phones, smart cameras, edge nodes, …)
    → ***focus: low memory and computational complexity during inference***

SotA in NN Compression

- typically three steps
- reduction of parameters, e.g.
  - eliminating neurons (pruning)
  - reducing the entropy of a tensor
  - decomposing/transforming a tensor
- reducing the precision of parameters (i.e. quantization)
- performing entropy coding



Han et al., ICLR 2016

Relation to Network Architecture Search (NAS)

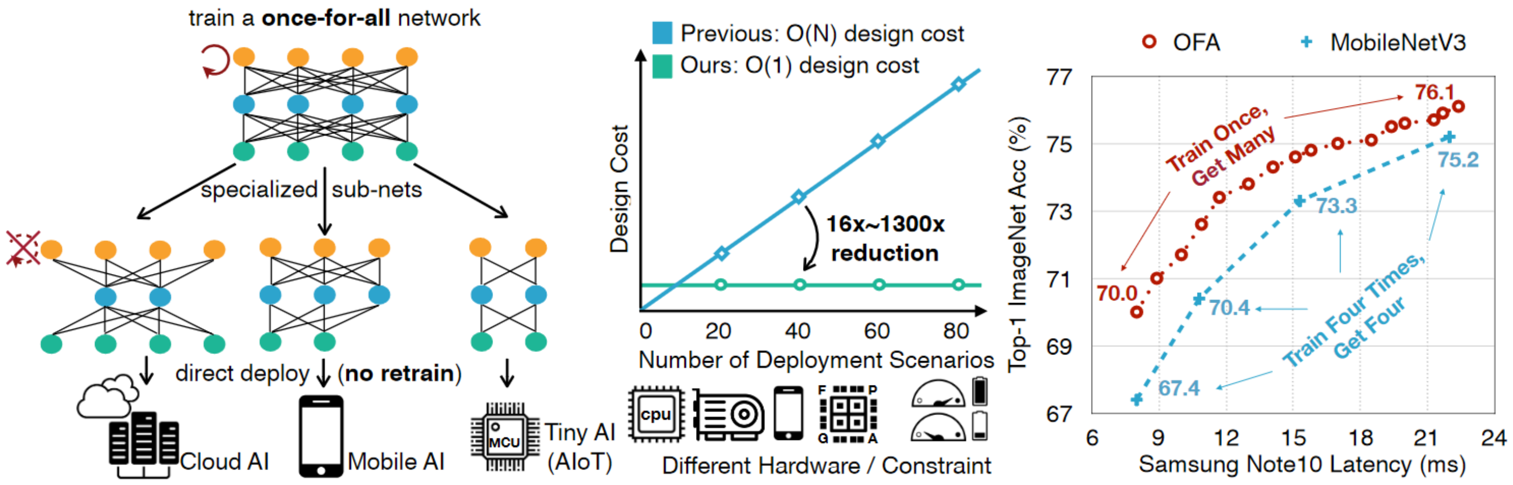
Model	Hardware	Power (W)	Hours	kWh·PUE	CO_2e
$\text{Transformer}_{base}$	P100x8	1415.78	12	27	26
Transformer_{big}	P100x8	1515.43	84	201	192
ELMo	P100x3	517.66	336	275	262
BERT_{base}	V100x64	12,041.51	79	1507	1438
BERT_{base}	TPUv2x16	—	96	—	—
NAS	P100x8	1515.43	274,120	656,347	626,155
NAS	TPUv2x1	—	32,623	—	—
GPT-2	TPUv3x32	—	168	—	—

[Strubell, et al. ACL 2019]

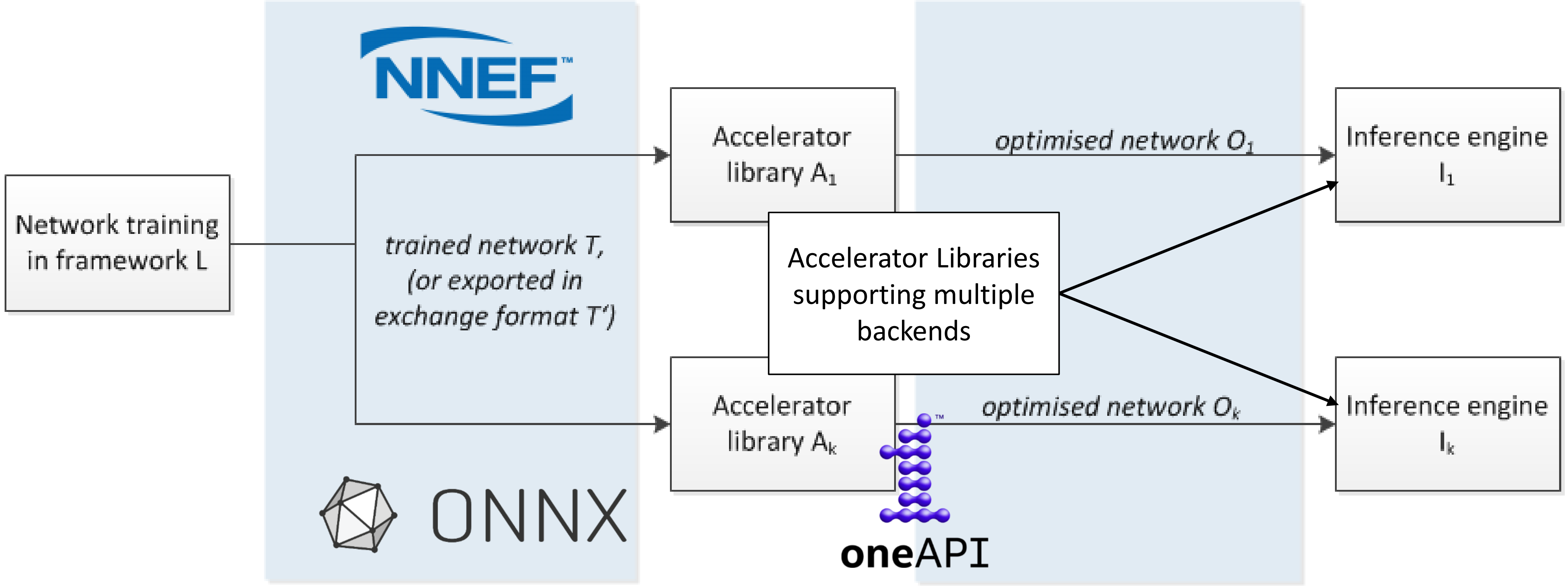
- finding an alternative architecture and train
  - architecture search is computationally expensive
  - training is then done using e.g. knowledge distillation, teacher-student learning
  - Faster NAS methods have been proposed, e.g. Single-Path Mobile AutoML [Stamoulis et al., IEEE JSTSP 2020]
  - needs also access to full training data, while fine-tuning could be done on partial data or application specific data

Relation to target hardware

- Optimising for target hardware
  - supported operations (e.g., sparse matrix multiplication, weight precisions)
  - relative costs of memory access and computing operations
- training a network, derive network for particular platform
  - first approaches [Cai, ICLR 2020]
  - no reliable prediction of inference costs on particular target architecture
  - in particular, prediction of speed and energy consumption
  - cf. autotune in inference of DL frameworks



Need for a standardised interface



standardised format is crucial, compression useful in some cases

compression is crucial

Standardization in MPEG (ISO/IEC JTC1 SC29)

- Develop interoperable compressed representation of neural networks
- Leverage the know-how in the MPEG on compression of various types of (multimedia) data
- Enable multimedia applications to benefit from the progress in machine learning using deep neural networks
- Cover a broad set of relevant use cases
  - selected image classification, visual content matching, content coding and audio classification as applications in which technology is validated

Design Considerations

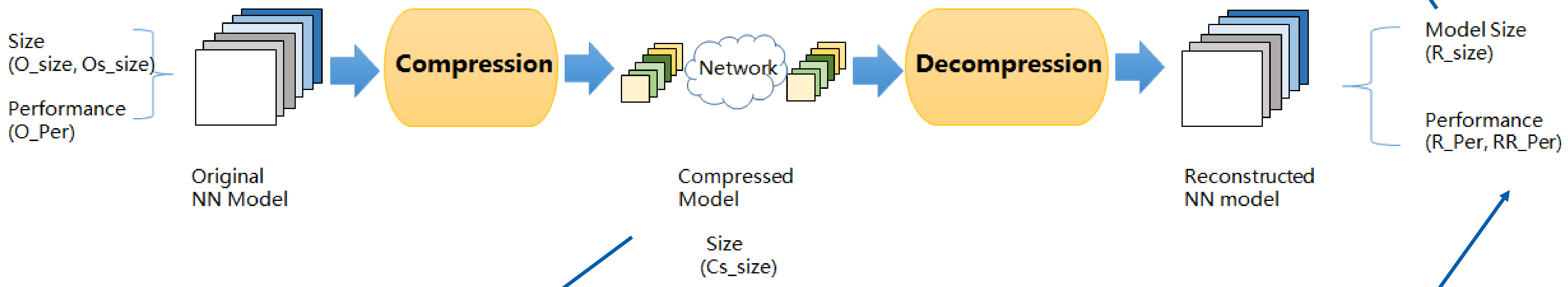
- Interoperability with exchange formats (NNEF, ONNX) and formats of common DL frameworks
- Reuse existing approaches for representing topology
- Agnostic to inference platform and its specificities
- Different types of networks, applications, … may be best served by different compression tools

Evaluating compression technologies

- Compression ratio
- Reconstruction of original parameters (cf. PSNR for multimedia data)
  - not a useful metric
  - performance in target application (e.g., image classification) needs to be measured (cf. perceptual quality metrics for multimedia)
  - requires models and data sets for each target application
- runtime/memory consumption
  - of the encoding/decoding process
  - inference using the resulting model

Evaluating compression technologies

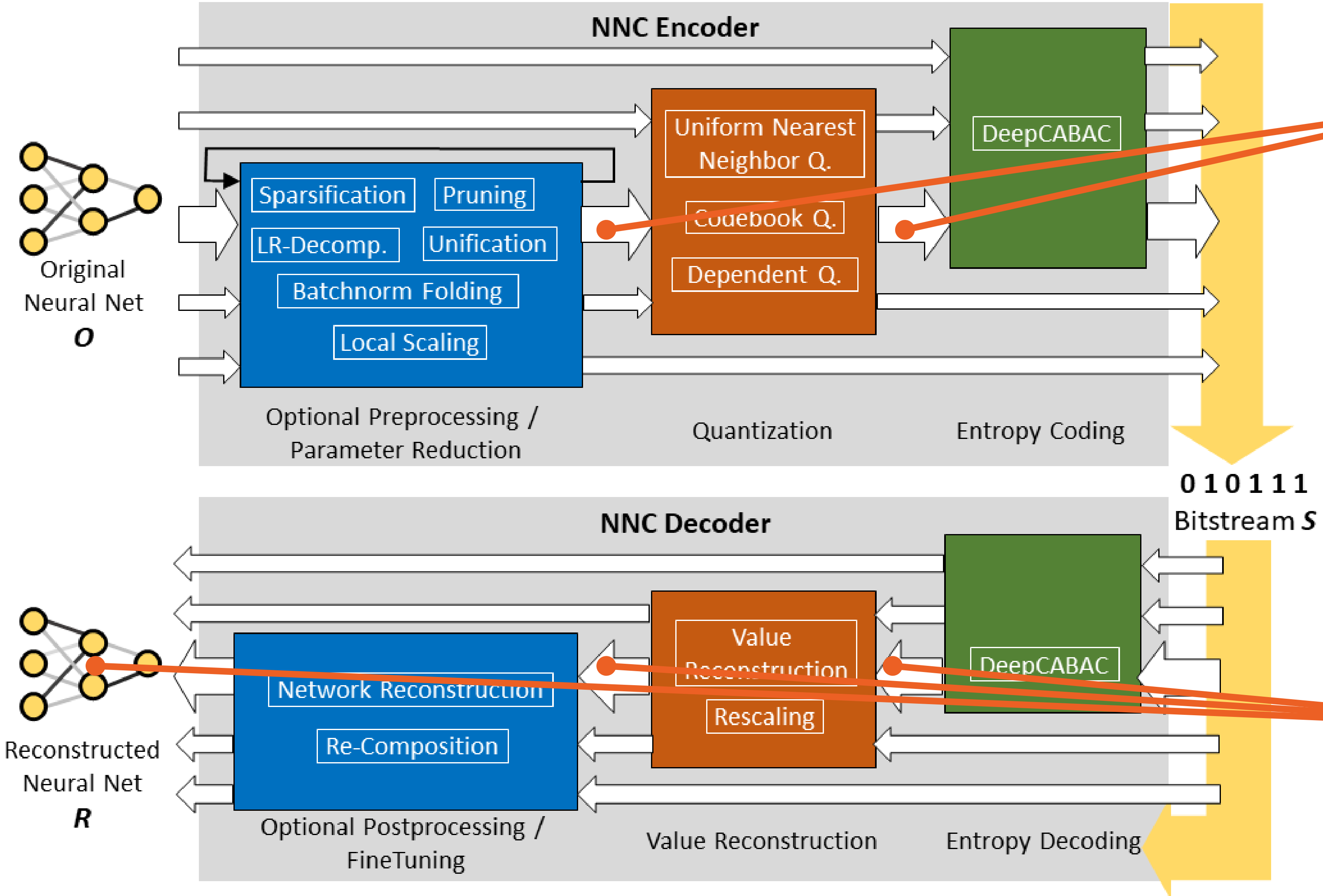
compression ratio of
model in memory
(depends on platform)



compression ratio of
stored/transmitted model
(format must be considered)

performance for task
(with/without finetuning)

Standard as a Toolbox



could be represented in any NN format

representation for storage/transmission

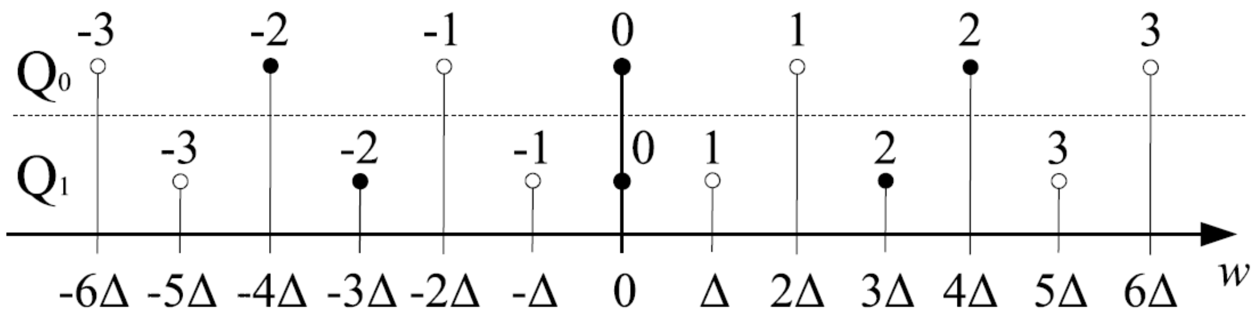
representation for inference

Parameter Reduction

- Sparsification
  - General sparsification
  - Micro-structured sparsification
- Pruning
  - estimate importance of weights to decide about pruning neurons
- Low-rank decomposition
  - approximate tensor as product of decomposition result (limiting number of parameters)
- Unification
  - generalisation of micro-structured sparsification (values other than 0)
- Batchnorm folding, local scaling
  - store batchnorm parameters, and apply to weights (better compressability of weights)
  - scaling factor per row (no additional parameters if used together with BN folding)

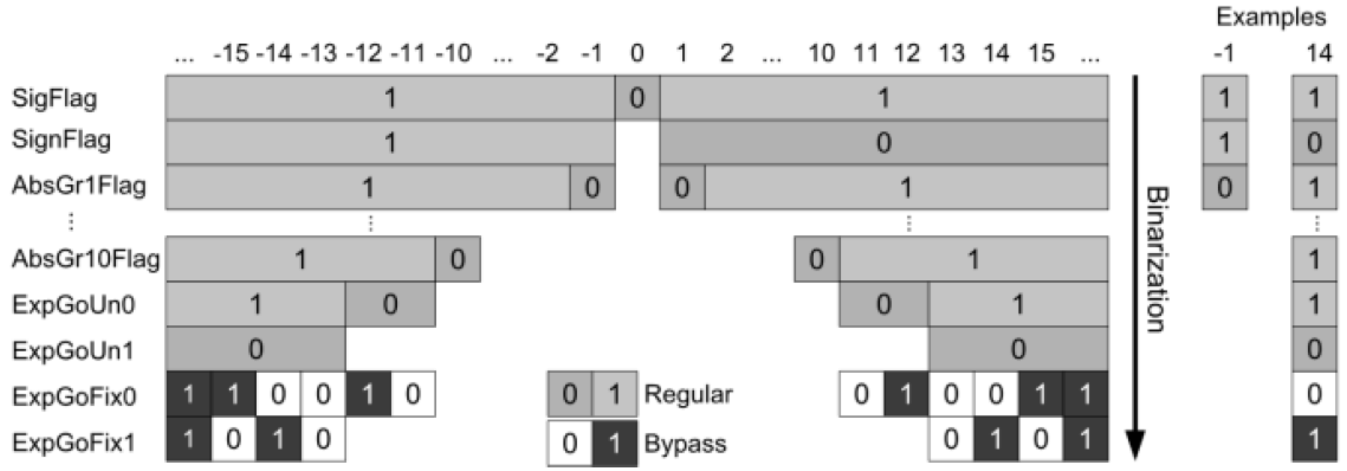
Quantisation

- Uniform Nearest Neighbor Quantization
- Codebook Quantization
- Dependent Scalar Quantization
  - Trellis-coded quantisation
  - two scalar quantisers, and procedure for switching between them (state-machine with 8 states)



Entropy Coding

- DeepCABAC
  - Adaptation of Context-adaptive Binary Arithmetic Coding (CABAC)
  - Binarization
  - Context-modelling
    - separate models for each of the flags
    - select from a fixed set of models
  - Arithmetic coding to regular and bypass bins



Decoding

- Output of encoded tensor
  - Integer or floating point
  - Block: set of combined tensors (e.g., components of decomposed tensor)
- Parallel decoding of parts of a tensor is supported
  - option to specify entry points for the parts during encoding

Interoperability with exchange formats

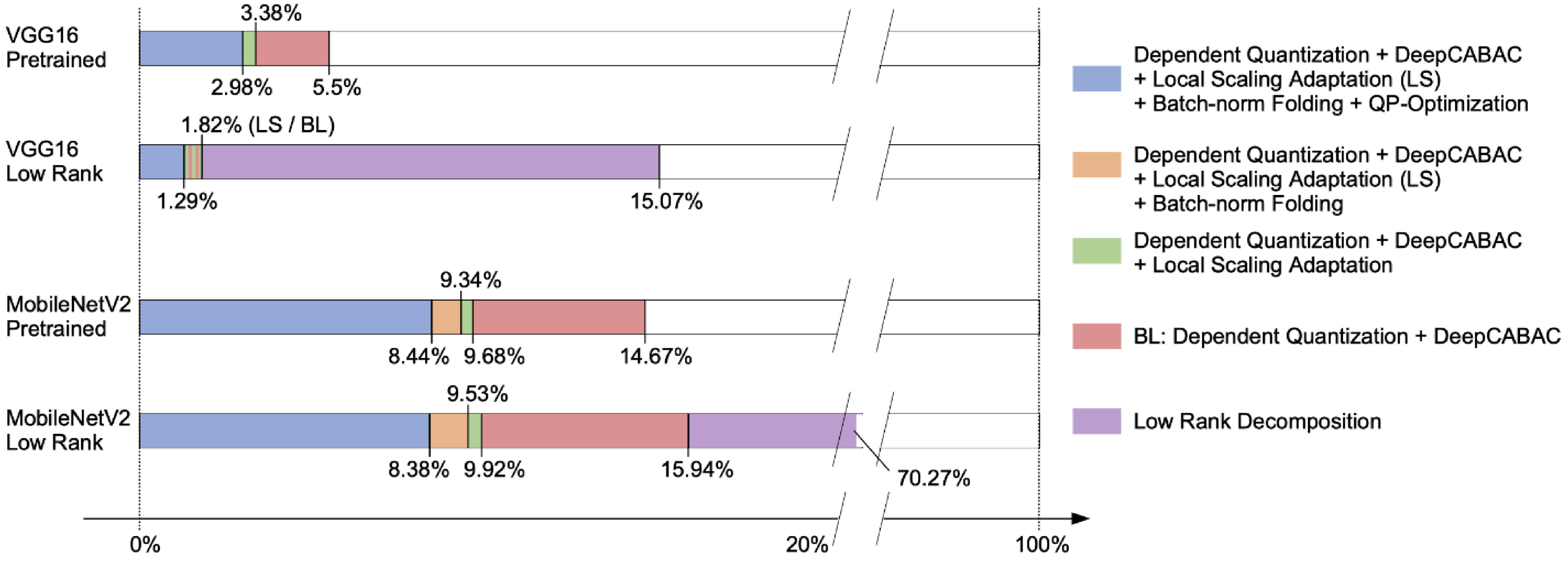
- Include network topology in encoded bitstream
  - ONNX, NNEF, Tensorflow, PyTorch
  - Supports encoding just some of the tensors
  - Compatibility with quantisation formats supported in those formats
- Include encoded tensors in exchange format
  - Recommended approach for NNEF and ONNX

Performance

NNR TRANSPARENT CODING RESULTS

Model	c_r in %	top-1 / top-5 acc. reconstr.	top-1 / top-5 acc. original	Orig. size (bytes)
VGG16	2.98	70.51 / 89.54	70.93 / 89.85	553.43 M
ResNet50	6.54	74.42 / 91.80	74.98 / 92.15	102.55 M
MobileNetV2	12.18	71.13 / 90.06	71.47 / 90.27	14.16 M
DCase	4.12	58.15 / 92.35	58.27 / 91.85	467.26 k
Model	**c_r in %**	**PSNR / SSIM reconstructed**	**PSNR / SSIM original**	**Orig. size (bytes)**
UC12B	17.34	29.98 / 0.954	30.13 / 0.956	304.72 k

Performance



Performance

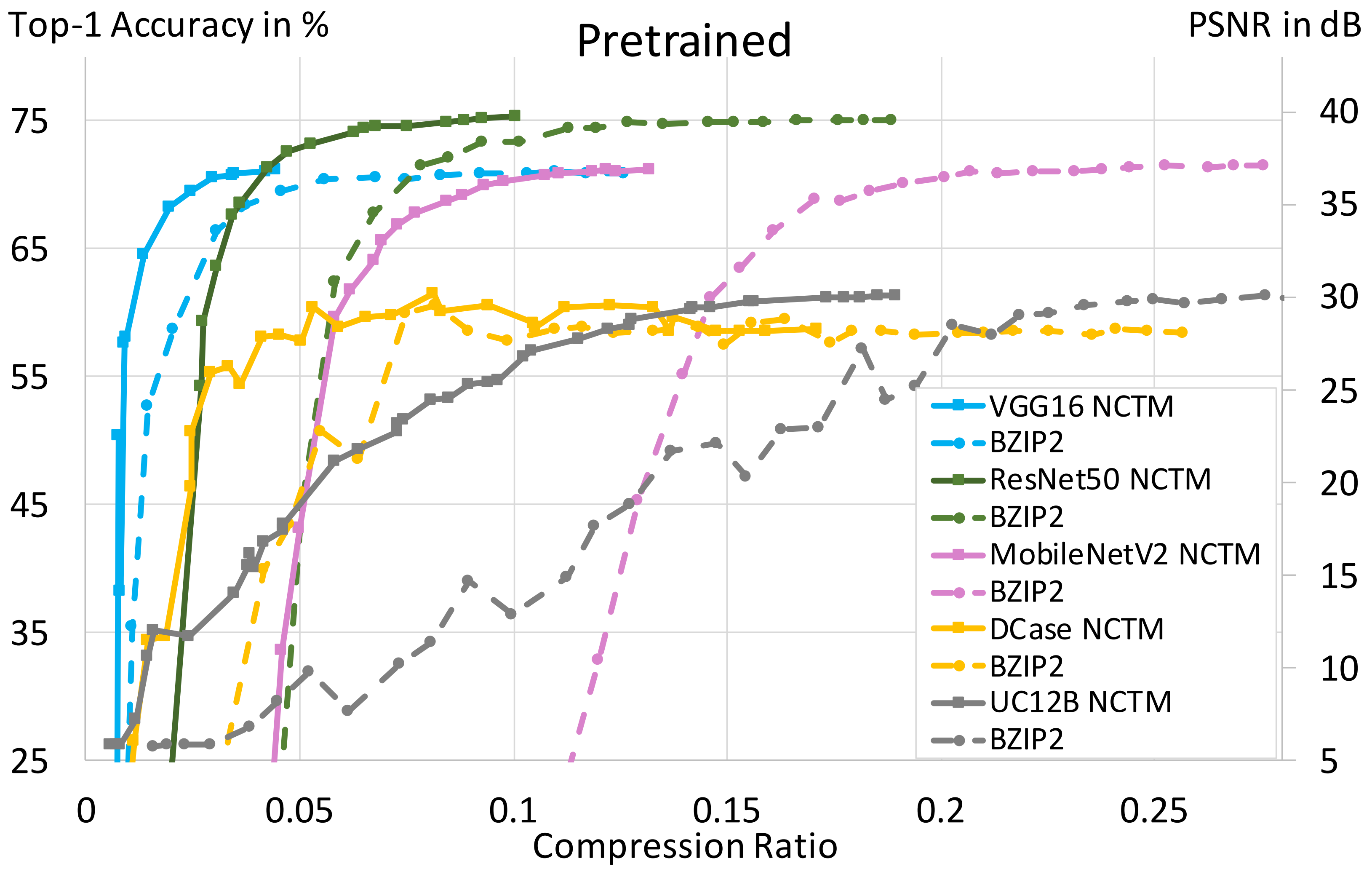


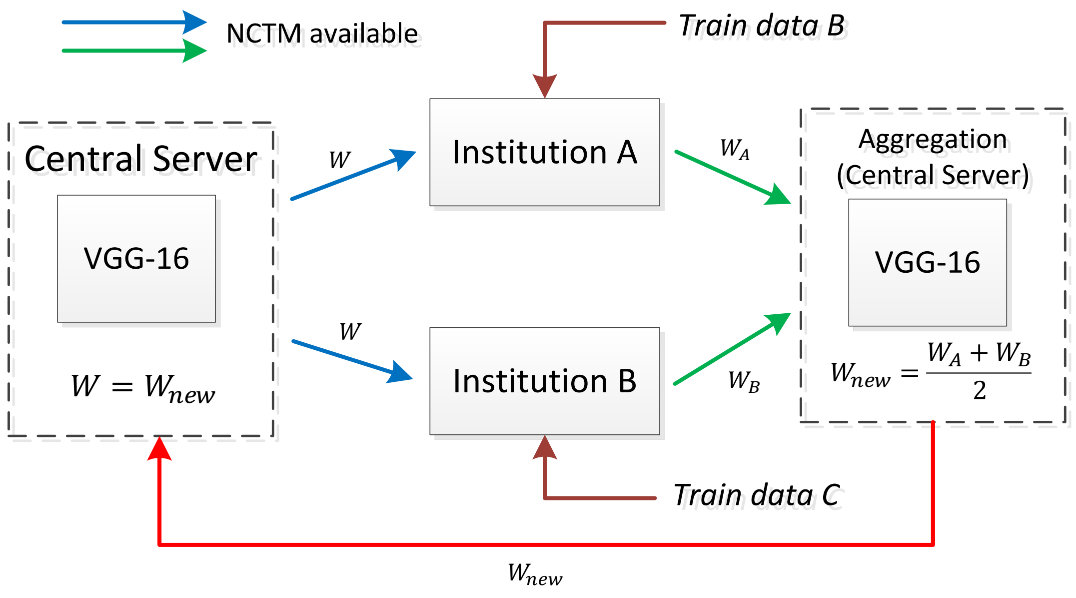
image classification

audio classification

image encoding

Ongoing work: incremental compression

- Use cases that need to send updated models
  - e.g. deploy to mobile devices, federated learning
- Encode model w.r.t. base model
  - support tensor updates and structural changes, e.g. transfer learning with different number of output classes
- Initial results
  - updates in distributed training can be represented at <1% of the base model size



Conclusion

- standard for compressing NN parameters
- compresses to less than 10% without performance loss
- interoperability with exchange formats
- status
  - compression standard (ISO/IEC 15938-17) going to FDIS ballot
  - reference software (ISO/IEC 15938-18) under CD ballot
  - work on incremental compression ongoing (to become 2nd ed. of pt. 17)

THE INNOVATION COMPANY

Werner Bailer
Co-chair of MPEG AhG on Neural Network Compression
werner.bailer@joanneum.at

This work has received funding from the European Union's Horizon 2020 research and innovation programme, under grant agreement no. 951911
AI4Media (https://ai4media.eu).

www.joanneum.at/digital