Something about Huffman Coding

1

References for Developing Huffman Codes with Better Properties:

- 1. Error States and **Synchronization Recovery** for Variable Length Codes, IEEE T-Commun., 783-792, Feb/March/April 1994
- 2. **Memory Efficient and High-Speed Search Huffman Coding**, IEEE T-Commun., pp. 2576-2581, Oct 1995 (Reza Hashemian)
- 3. **Synchronization Recovery of variable-Length Codes**, IEEE T-IT, pp. 219-227, Jan. 2002
- 4. **Verification of Minimum-Redundancy Prefix Codes**, IEEE T-IT, pp. 1399-1404, April 2006
- 5. **The Construction of Variable Length Codes with Good Synchronization Properties**, IEEE T-IT, pp. 1696-1700, April 2009.

2

6. **Huffman Codes for Arbitrary-Side Growing Trees**:
- **Memory Efficient Hierarchical Lookup Tables for Massive Arbitrary-Side Growing Huffman Tree Decoding**, IEEE Trans. on Circuits and Systems for Video Technology, Vol. 18, pp. 1335-1346, Oct. 2008.

7. **Reversible Variable Length Codes**- Huffman code based Approach:
- **On Constructing the Huffman-code-based Reversible Variable-Length Codes**, IEEE T-Comm., Sept. 2001.
- **Modified Symmetrical Reversible Variable-Length Code and Its Theoretical Bounds**, IEEE T-IT, Sept. 2001.
- **On Error Detection and Error Synchronization of Reversible Variable Length Codes**, IEEE T-Comm., May 2005.
- **Two Algorithms for Construction Efficient Huffman-Code based Reversible Variable Length Codes**, IEEE T-Comm., Jan, 2008.

3

8. A different View for Constructing Huffman Codes,
- " **Windowed Huffman Coding Algorithm with Size Adaptation**," IEE Proceeding-I, pp. 109-113, April 1993.

9. How to Construct an efficient **adaptive Huffman Codes** for dealing with Streaming Data? → **Dynamic Huffman Codes**

10. If we switch the codewords with equal length, the compression ratio will be preserved. Can you build a **secure Huffman code** based on this fact?

11. What is the effect of switching the equal-probable combined (merged) symbol and an original symbol, during the code-tree construction?

→ **Variance of the codeword length matters**!

4

Alistair Moffat, "Huffman Coding," ACM Computing Surveys 52(4):1-35, DOI: 10.1145/3342555, August **2019**.

This article presents a tutorial on Huffman coding and surveys some of the developments that have flowed as a consequence of Huffman's original discovery, including details of code calculation and of encoding and decoding operations.
The author also surveys related mechanisms, covering both **arithmetic coding** and the recently developed **asymmetric numeral systems (ANS)** approach and briefly discuss other Huffman-coding variants, including **length-limited codes**.

5

Open Access Article

A Review of the Asymmetric Numeral System and Its Applications to Digital Images

Entropy **2022**, *24*(3), 375; https://doi.org/10.3390/e24030375

by Ping Ang Hsieh [1] and Ja-Ling Wu [1,2,*]

[1] Graduate Institute of Networking and Multimedia, Taipei 10617, Taiwan
[2] Department of Computer Science and Information Engineering, National Taiwan University, Taipei 10617, Taiwan

ANS is one of the most recently proposed **entropy coding** methods. **Fast execution speed** and **close to the theoretical limit compression performance** are the prominent features of ANS; therefore, it has been **primarily adopted by industrials**. **Jarek Duda** first proposed ANS in 2007, and it was adopted and implemented by **Facebook** in 2015, namely, Zstandard, which is **open-sourced** and used in various fields such as **Linux Kernel /Hadoop /Mysql/ FreeBSD**. **Apple** also released its ANS implementation—LZFSE [5]—in 2015 and used it at the bottom layer **of iOS and macOS**. **Google** launched its lossless compression standard—pik—in 2019, in which the entropy coding part also uses ANS. **Microsoft** also applied for ANS-related patents in 2019. In addition to the industry giants mentioned above, the **JPEG standard committee** began drafting the new compression standard **JPEG XL** in 2017. ANS also plays a significant role in its entropy coding. We can see that in the past five years, ANS has been widely accepted and adopted by the IT giants, but in the compression academia community and nonexpert IT industry, the awareness and the adoption of ANS for Multimedia compression is still in its infancy.

6

References for traditional Transform Coding and DCT

1. Fast Discrete Cosine Transform Algorithms - Website of Project Nayuki
http://www.nayuki.io/page/fast-discrete-cosine-transform-algorithms
Source Codes written in Java, Python, C, C++, …, are provided.

2. Fast DCT algorithm: math derivation
http://fourier.eng.hmc.edu/e161/lectures/dct/node2.html

3. Introduction to Transform Coding
http://www.cmlab.csie.ntu.edu.tw/cml/dsp/training/coding/transform/intro.html
both 1-D and 2-D transforms are included

4. Vivek K. Goyal, "Theoretical Foundations of Transform Coding," IEEE Signal Processing Magazine, Sept 2001.

5. Efficient Fixed-Point Approximations of the 8x8 Inverse Discrete Cosine Transform,
http://www.reznik.org/papers/SPIE07_MPEG-C_IDCT.pdf

6. The Berkeley Software MPEG-1 Video Decoder, https://dl.acm.org/doi/pdf/10.1145/1047936.1047944

7

The Recent Interaction between Information Theory (Data Compression) and DNN

8

References

Neural Network Compression

S Wiedemann, H Kirchhoffer, S Matlage, P Haase, A Marban, T Marinc, D Neumann, T Nguyen, A Osman, H Schwarz, D Marpe, T Wiegand, W Samek. DeepCABAC: A Universal Compression Algorithm for Deep Neural Networks. *IEEE Journal of Selected Topics in Signal Processing*, 14(4):700-714, 2020.
`http://dx.doi.org/10.1109/JSTSP.2020.2969554`

S Yeom, P Seegerer, S Lapuschkin, S Wiedemann, KR Müller, W Samek. Pruning by Explaining: A Novel Criterion for Deep Neural Network Pruning. *arXiv:1912.08881*, 2019.
`https://arxiv.org/abs/1912.08881`

S Wiedemann, H Kirchhoffer, S Matlage, P Haase, A Marban, T Marinc, D Neumann, A Osman, D Marpe, H Schwarz, T Wiegand, W Samek. DeepCABAC: Context-adaptive binary arithmetic coding for deep neural network compression. *Joint ICML'19 Workshop on On-Device Machine Learning & Compact Deep Neural Network Representations (ODML-CDNNR)*, 1-4, 2019. ***** Best paper award *****
`https://arxiv.org/abs/1905.08318`

S Wiedemann, A Marban, KR Müller, W Samek. Entropy-Constrained Training of Deep Neural Networks. *Proceedings of the IEEE International Joint Conference on Neural Networks (IJCNN)*, 1-8, 2019.
`http://dx.doi.org/10.1109/IJCNN.2019.8852119`

9

Efficient Deep Learning

S Wiedemann, KR Müller, W Samek. Compact and Computationally Efficient Representation of Deep Neural Networks. *IEEE Transactions on Neural Networks and Learning Systems*, 31(3):772-785, 2020.
`http://dx.doi.org/10.1109/TNNLS.2019.2910073`

S Wiedemann, T Mehari, K Kepp, W Samek. Dithered backprop: A sparse and quantized backpropagation algorithm for more efficient deep neural network training. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 3096-3104*, 2020.
`https://dx.doi.org/10.1109/CVPRW50498.2020.00368`

A Marban, D Becking, S Wiedemann, W Samek. Learning Sparse & Ternary Neural Networks with Entropy-Constrained Trained Ternarization (EC2T). *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 3105-3113*, 2020.
`https://dx.doi.org/10.1109/CVPRW50498.2020.00369`

1. **Entropy-Constrained Training of Deep Neural Networks**
Simon Wiedemann, et al. Available at: arXiv:1812.07520v2 [cs.LG] 19 Dec 2018
2. **MPEG-7 part 17**: Standard on "Compression on Neural Networks for Multimedia Content Description and Analysis" : **Overview of the Neural Network Compression and Representation (NNR) Standard**
IEEE T-CSVT, VOL. 32, NO. 5, pp. 3203-3216, MAY 2022

10

- Neural Network Coding and Representation (NNR) is the first international standard for efficient compression of neural networks (NNs). The standard is designed as a toolbox of compression methods, which can be used to create coding pipelines. It can be either used as an independent coding framework (with its own bitstream format) or together with external neural network formats and frameworks. For providing the highest degree of flexibility, the network compression methods operate per parameter tensor in order to always ensure proper decoding, even if no structure information is provided.
- The NNR standard contains compression-efficient quantization and deep context-adaptive binary arithmetic coding (DeepCABAC) as core encoding and decoding technologies, as well as neural network parameter pre-processing methods like sparsification, pruning, low-rank decomposition, unification, local scaling and batch norm folding.
- NNR achieves a compression efficiency of more than 97% for transparent coding cases, i.e. without degrading classification quality, such as top-1 or top-5 accuracies. This paper provides an overview of the technical features and characteristics of NNR.

11