Lossless Multi-Component Image Compression based on Integer Wavelet Coefficient Prediction using Convolutional Neural Networks

Eze L. Ahanonu[1], Michael W. Marcellin[1], and Ali Bilgin[1, 2]

[1]Department of Electrical and Computer Engineering, [2]Department of Biomedical Engineering, The University of Arizona, Tucson, AZ, USA

Abstract

- This work extends the methods proposed in [1], termed *Wavelet Prediction Compression (WPC),* for lossless multi-component image compression. The extensions proposed here are referred to as *Multi-Component Wavelet Prediction Compression (MCWPC).* The potential benefits of both inter- and intra-component prediction are considered. A greedy search procedure is proposed in order to allow efficient construction of prediction models. An end-to-end encoder/decoder is implemented and the resulting bitrate are compared with current methods.

Proposed Model

- During encoding (Figure 1), an M-component input image is first subject to a DC-shift and (optional) Reversible Color Transform (RCT).
- The resulting image then undergoes a single level decomposition using a reversible (integer) Discrete Wavelet Transform (DWT) [2] independently across channels to obtain the set of subbands $\mathcal{D}_1^m = \{LL_1^m, HL_1^m, LH_1^m, HH_1^m\}$, $m = 1, \dots, M$ (e.g. $M = 3$ for RGB/YUV images).
- Each subband is taken as input into a CNN encoding model (CNN_{enc_1}) to produce the detail subband predictions $\widehat{\mathcal{D}}_1^m = \{\widehat{HL}_1^m, \widehat{LH}_1^m, \widehat{HH}_1^m\}$.
- A set of subband residuals is calculated as $\overline{\mathcal{D}}_1^m = \{\overline{HL}_1^m, \overline{LH}_1^m, \overline{HH}_1^m\} = \{HL_1^m - \widehat{HL}_1^m, LH_1^m - \widehat{LH}_1^m, HH_1^m - \widehat{HH}_1^m\}$.
- The DWT decomposition and prediction procedure is recursively repeated on the approximation subbands $LL_n^1, LL_n^2, \dots, LL_n^m$ for a user-defined N decompositions.
- After the final decomposition, the original image samples are represented in terms of the residual subband sets along with the final approximation subbands: $\{\overline{\mathcal{D}}_1^1, \dots, \overline{\mathcal{D}}_1^M, \overline{\mathcal{D}}_2^1, \dots, \overline{\mathcal{D}}_N^M, LL_N^1, \dots, LL_N^M\}$.
- Both the original and residual subbands are divided into 64x64 blocks. The first order entropy is calculated for each block in both, and the block with the lowest entropy (between original and residual) is used in the final codestream. A 1-bit flag is sent with each block to notify the decoder which block was used.
- Approximation, detail, and residual coefficients are entropy coded to obtain a final codestream.
- Decoding involves reproducing predictions with identical CNNs and summing with decoded residuals for perfect coefficient reconstruction.

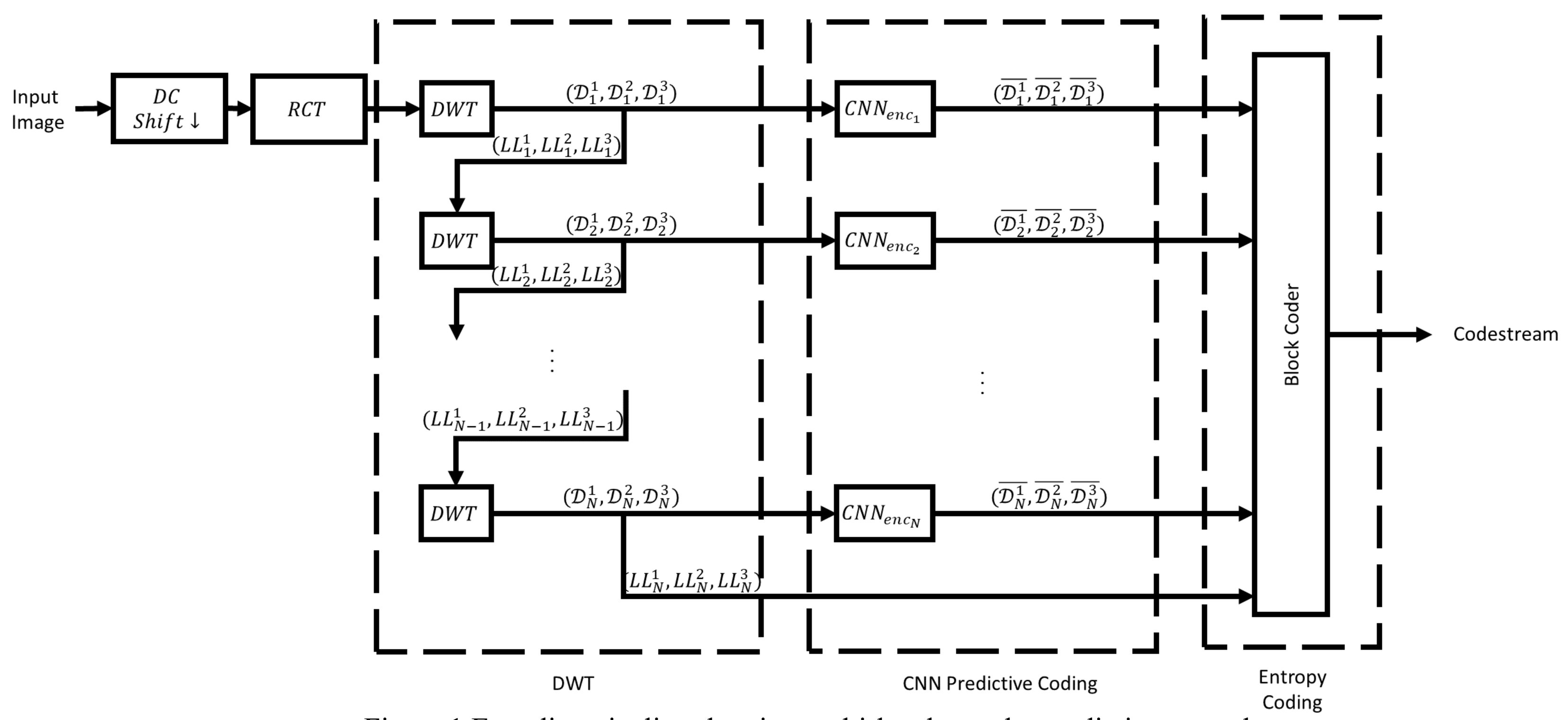

Figure 1 Encoding pipeline showing multi-level wavelet prediction procedure

Construction of Prediction Models

- The prediction strategy in MCWPC is the same as for WPC: maximize decorrelation by feeding the maximum number of allowable subbands (which preserve causality) into each CNN at each prediction step.
- Consideration of both inter- and intra-component prediction in MCWPC results in a significantly larger space of possible prediction configurations which can be considered for an optimal prediction model.
- A greedy search algorithm (Algorithm 1) is proposed to more efficiently construct prediction models which produce prediction residuals that maximize potential bit-rate reductions compared to coding of original coefficients.
- In Algorithm 1, $(\mathcal{S}_n - s_j)$ denotes the set of all subbands in $\mathcal{S}_n$ except for s_j.
- After completing the search, the set $\mathcal{M}_n$ will contain CNN predictors in the order in which they should be applied during encoding, and describes how CNN_{enc_n} should be implemented.

For the set of subbands across all components at a given decomposition level,

$$\mathcal{S}_n = \{LL_n^1, \dots, LL_n^M, HL_n^1, \dots, HL_n^M, LH_n^1, \dots, LH_n^M, HH_n^1, \dots, HH_n^M\}$$

and the (initially empty) ordered set of CNN predictions $\mathcal{M}_n = \{\}$, the search procedure proceeds as follows:

1. For each subband s_j in $\mathcal{S}_n$, train the model $(\mathcal{S}_n - s_j) \rightarrow s_j,\ j = 1, \dots, |\mathcal{S}_n|$.
2. Find the model $(\mathcal{S}_n - s_j^*) \rightarrow s_j^*$ which maximizes entropy reduction over the test dataset.
3. Append the model $(\mathcal{S}_n - s_j^*) \rightarrow s_j^*$ to the end of $\mathcal{M}_n$.
4. Remove s_j from $\mathcal{S}_n$.
5. If $\mathcal{S}_i$ still contains detail subbands, return to (1). Otherwise, terminate the search.

Algorithm 1 Greedy search procedure for subband prediction models

Construction of Prediction Models (Cont'd)

- The feasibility of the Algorithm 1 is demonstrated by constructing prediction models (individually) for 3-component RGB and YUV images for a single level of DWT decomposition.
- The CNNs trained are comprised of 10 convolutional layers, with each layer containing 64 3x3 filters and ReLU activation (with the exception of the output layer).
- Networks are trained in TensorFlow [3] for 100 epochs using the Adam optimizer with a batch size of 32, and fixed learning rate and L2-regularization of 1e-4. An MSE loss function is used to evaluate network output.
- Training/validation/testing data is sourced from a subset of 2000 (1000/500/500 split) 2048x2048 8-bit color images from the RAISE high-resolution raw image dataset [4].
- Because the contribution of bits for each subband within the final codestream is non-uniform, the entropy reductions computed during the search must be weighted by their estimated relative contributions. These weight are given in Table 1 and are computed over the test dataset from JPEG2000 codestreams generated using OpenJPEG [5].

	HL_1	LH_1	HH_1
R	0.079	0.080	0.068
G	0.080	0.081	0.069
B	0.078	0.079	0.067

(a)

	HL_1	LH_1	HH_1
Y	0.112	0.114	0.096
U	0.063	0.062	0.066
V	0.054	0.053	0.059

(b)

Table 1 Relative contribution of (a) RGB and (b) YUV subbands to full JPEG2000 codestream

- The weighted entropy reduction at each iteration of the Algorithm 1 for RGB and YUV subband sets at the first DWT decomposition level are given in Tables 2 and 3, respectively.
- The subband which achieved the largest reduction in entropy at a given iteration are shown in bold.
- The search was terminated when entropy reductions of all remaining subband fell below 0.005, at which point only negligible rate-reductions would be achieved by additional prediction steps.

Iteration:	1	2	3	4	5	6	7	8	9
R_{HL}	0.42	**0.42**							
R_{LH}	**0.45**								
R_{HH}	0.27	0.27	0.21	0.21	0.21	**0.21**			
G_{HL}	0.42	0.39	0.36	0.12	0.12	0.12	**0.06**		
G_{LH}	0.42	0.42	0.39	**0.39**					
G_{HH}	0.21	0.24	0.24	0.24	0.24	0.21	0.12	0.06	0
B_{HL}	0.39	0.39	**0.39**						
B_{LH}	0.45	0.39	0.36	0.39	0.12	0.12	0.09	**0.09**	
B_{HH}	0.24	0.24	0.24	0.24	**0.24**				

Table 2 Entropy reduction across search iterations in bpp (RGB)

Iteration:	1	2	3	4	5	6	7
Y_{HL}	**0.24**						
Y_{LH}	0.24	**0.18**					
Y_{HH}	0.12	0.09	**0.06**				
U_{HL}	0.06	0.03	0.03	**0.03**			
U_{LH}	0.06	0.06	0.03	0.03	0.03	**0.03**	
U_{HH}	0.06	0.06	0.03	0.03	0	0	0
V_{HL}	0.06	0.03	0.03	0.03	0.03	0	0
V_{LH}	0.06	0.03	0.03	0.03	**0.03**		
V_{HH}	0.03	0.03	0.03	0.03	0.03	0	0

Table 3 Entropy reduction across search iterations in bpp (YUV)

- The final prediction models for the first decomposition level of RGB and YUV images were

$$CNN_{enc_1}^{RGB} = \{R_{LH_1}, R_{HL_1}, B_{HL_1}, G_{LH_1}, B_{HH_1}, R_{HH_1}, G_{HL_1}, B_{LH_1}\}$$

and

$$CNN_{enc_1}^{YUV} = \{Y_{HL_1}, Y_{LH_1}, Y_{HH_1}, U_{HL_1}, V_{LH_1}, U_{LH_1}\}$$

respectively.

- For the YUV model, the choice of initially predicting luminance (Y) subbands may be considered optimal by observing that the total weighed entropy reduction for UV subbands at the first search iteration is 0.33bpp, which is less than the 0.48bpp reduction that is achieved through causal prediction of Y subbands.
- The overall estimated bitrate reductions for YUV components after prediction is 0.57bpp.
- The RGB search yielded larger entropy reductions, and a slower decay in prediction performance over search iterations than were observed for the YUV case.
- These increased reductions may be attributed to the stronger correlations which exists among RGB components, compared to the YUV components.
- While we observe a more substantial 2.25bpp bitrate reduction using the RGB prediction model, images compressed in RGB space suffer a 3.1bpp average bitrate increase compared to YUV compressed images.

Compression Experiments

- Using the final prediction models, an end-to-end encoder/decoder is implemented.
- Compressed codestreams are generated by supplying prediction residuals to the context-based binary arithmetic coder used in OpenJPEG [5]. Though OpenJPEG produces JPEG2000 compliant codestreams, our modified version is not JPEG2000 compliant. Here, OpenJPEG is only used for its arithmetic coder and its ability to produce the final compressed file by combining compressed data from each codeblock.
- For comparison, all images are also compressed using Lossless JPEG2000 [8] (with and without RCT), JPEG-LS [6], and FLIF [7].
- Images are additionally compressed using methods from WPC [1] independently on each component to determine bitrate reductions which may be attributed to considering cross-component dependencies. These results are given in Table 4.

Method	Average Bitrate (bpp)
JPEG-LS	10.69
FLIF	6.73
Lossless JPEG2000 (YUV)	7.77
Lossless JPEG2000 (RGB)	10.96
WPC (YUV)	7.36
MCWPC (YUV)	7.18
MCPWC (RGB)	8.35

Table 4 Average bitrate achieved by the proposed method compared with other lossless compression methods on the test dataset

- MCWPC (YUV) achieves a 7.6% bitrate reduction over Lossless JPEG2000 (YUV).
- MCWPC (RGB) achieves a 23.8% bitrate reduction over Lossless JPEG2000 (RGB), but is not able to achieve substantial enough bitrates to outperform MCWPC (YUV), let alone Lossless JPEG2000 (YUV).
- WPC (YUV) provides a 5.3% bitrate reduction over Lossless JPEG2000 (YUV).
- In comparing WPC (YUV) to MCWPC (YUV), we see a 2.5% bitrate reduction is attributed to exploiting cross-component dependencies. This indicates that, while intra-component dependencies are more significant, exploiting inter-component dependencies provides a modest increase in compression performance.
- Figure 2 compares the bitrate achieved by other lossless methods compared to that achieved by MCWPC (YUV) for each test image.
- From Figure 2 it is evident that MCWPC achieves its strongest performance at higher rates. This is due to high bitrate images having higher energy detail subbands, which when compared to lower energy prediction residuals produces significant rate-reductions.
- Conversely, low bitrate images have little content in detail subbands for the CNN model to predict, leading to diminished rate reductions.
- When compared to the results in [1], MCWPC achieves a less comparative performance with FLIF than WPC. This indicates that FLIF is able to exploit cross-component redundancies in a way that MCWPC is not yet able to achieve, resulting in a performance gap when moving from the grayscale to multi-component regime.

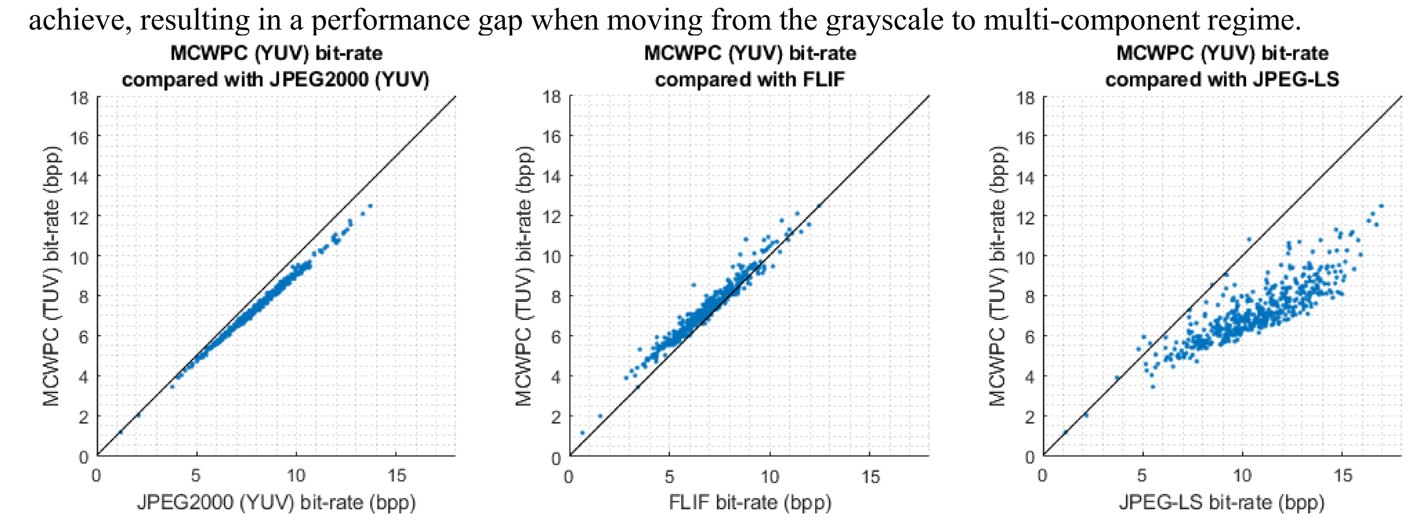

Figure 2 Bitrate achieved by various methods compared to MCWPC, where each datapoint represents a single test image

Conclusions

- This work considered the extension of WPC to multi-component image compression. A greedy search approach was proposed to reduce the number of models which must be considered to produce a good prediction framework. An end-to-end encoder/decoder was implemented, and the final rates were compared with existing methods.

References

[1] Eze Ahanonu, "Lossless image compression using reversible integer wavelet transforms and convolutional neural networks," Master's Thesis, 2019.
[2] A. Calderbank, I. Daubechies, W. Sweldens and B.-L. Yeo, "Wavelet Transforms That Map Integers to Integers," Applied and Computational Harmonic Analysis, vol. 5, no. 3, pp. 332-369, 1998.
[3] M. Abadi, "TensorFlow: a system for large-scale machine learning," In Proceedings of the 12th USENIX conference on Operating Systems Design and Implementation (OSDI'16). USENIX Association, Berkeley, CA, USA, 265-283.
[4] D.-T. Dang-Nguyen, C. Pasquini, V. Conotter and G. Boato, "RAISE A Raw Images Dataset for Digital Image Forensics," 18-20 March 2015. [Online]
[5] Université de Louvain, "OpenJPEG," 2019. [Online]. Available: http://www.openjpeg.org/.
[6] M. J.Weinberger, G. Seroussi, and G. Sapiro, "The LOCO-I lossless image compression algorithm: Principles and standardization into JPEG-LS," Trans. Img. Proc., vol. 9, no. 8, pp. 1309-1324, Aug. 2000.
[7] Jon Sneyers and Pieter Wuille, "FLIF: free lossless image format based on MANIAC compression," in 2016 IEEE International Conference on Image Processing, ICIP 2016, Phoenix, AZ, USA, September 25-28, 2016, 2016, pp. 66-70.
[8] David Taubman and Michael Marcellin, JPEG2000 Image Compression Fundamentals, Standards and Practice, Springer Publishing Company, Incorporated, 2002.