POLITECNICO DI TORINO
Repository ISTITUZIONALE

Hybrid Recurrent-Attentive Neural Network for Onboard Predictive Hyperspectral Image Compression

Original
Hybrid Recurrent-Attentive Neural Network for Onboard Predictive Hyperspectral Image Compression / Valsesia, Diego; Bianchi, Tiziano; Magli, Enrico. - ELETTRONICO. - (2024), pp. 7898-7902. (Intervento presentato al convegno IGARSS 2024 - 2024 IEEE International Geoscience and Remote Sensing Symposium tenutosi a Athens (Greece) nel 07-12 July 2024) [10.1109/igarss53475.2024.10641584].

Availability:
This version is available at: 11583/2992848 since: 2024-09-27T12:20:20Z

Publisher:
IEEE

Published
DOI:10.1109/igarss53475.2024.10641584

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright
IEEE postprint/Author's Accepted Manuscript

©2024 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collecting works, for resale or lists, or reuse of any copyrighted component of this work in other works.

(Article begins on next page)

10 November 2024

HYBRID RECURRENT-ATTENTIVE NEURAL NETWORK FOR ONBOARD PREDICTIVE HYPERSPECTRAL IMAGE COMPRESSION

Diego Valsesia, Tiziano Bianchi, Enrico Magli

Politecnico di Torino
Department of Electronics and Telecommunications
Torino, Italy

ABSTRACT

AI-based compression is gaining popularity for traditional photos and videos. However, such techniques do not typically scale well to the task of compressing hyperspectral images, and may have computational requirements in terms of memory usage and total floating point operations that are prohibitive for usage onboard of satellites. In this paper, we explore the design of a predictive compression method based on a novel neural network design, called LineRWKV. Our neural network predictor works in a line-by-line fashion limiting memory and computational requirements thanks to a recurrent inference mechanism. However, in contrast to classic recurrent networks, it relies on an attention operation that can be parallelized for training, akin to Transformers, unlocking efficient training on large datasets, which is critical to learn complex predictors. In our preliminary results, we show that LineRWKV significantly outperforms the state-of-the-art CCSDS-123 standard and has competitive throughput.

***Index Terms*—** AI compression, predictive coding, attention, RWKV.

1. INTRODUCTION

Satellite hyperspectral imagery plays a pivotal role in numerous fields, serving as a valuable tool for Earth observation and analysis thanks to the ability of capturing a vast range of wavelengths across the electromagnetic spectrum. However, the ever-growing spatial and spectral resolution of hyperspectral imagers can produce tremendous amounts of data to be transmitted to the ground segment, making compression a pivotal component of modern missions. Traditionally, onboard hyperspectral image compression has been addressed either with a transform coding paradigm or with a predictive coding paradigm. The latter produced the most recent and highly-successful CCSDS-123 standard [1] which employs an adaptive spatio-spectral filter to perform pixel prediction. Recently, there is growing interest towards AI-based compression, as it has showcased strong performance on traditional photos and videos. However, significant challenges still hinder its usage for onboard hypespectral compression. In particular, the complexity of neural-network approaches tends to be rather high and further exacerbated by the high memory requirements posed by processing hyperspectral data cubes.

Existing approaches follow the auto-encoder design of neural networks for compression [2], where an encoder neural network generates a compact latent space that can be quantized and entropy-coded serving as the compressed data, which are decodable by a decoder neural network. This approach, reminiscent of transform coding, is extremely effective as it allows rate-distortion optimized training and can, in principle, use neural network designs that best capture all the correlation patterns existing in the input image. However, this typically results in high computational and memory requirements due to the need to process the entire input hyperspectral cube into a compact code [3, 4]. Some works [5] attempt at reducing the complexity by carefully designing neural network operations and working on small groups of bands, but the result typically does not scale well to the high-quality, high-rate setting which is the most desirable for a real mission.

In this paper, we follow a different approach, i.e., we design a non-linear predictor for predictive coding, in the form of neural network with a special architecture, namely that can work in a recursive line-by-line manner. Processing a single line with all its bands at a given time can intuitively limit the memory and computational requirements of the model, as well as enabling continuous operation if synchronized with the acquisition by pushbroom sensors. In order to accomplish our goal, we borrow ideas from the natural language literature (NLP). In NLP, Transformers [6] have supplanted recurrent neural networks (RNNs), which would have been the most obvious implementation of a recursive model, thanks to the fact that Transformers can be trained in a parallel fash-

This study was carried out within the FAIR - Future Artificial Intelligence Research and received funding from the European Union Next-GenerationEU (PIANO NAZIONALE DI RIPRESA E RESILIENZA (PNRR) – MISSIONE 4 COMPONENTE 2, INVESTIMENTO 1.3 – D.D. 1555 11/10/2022, PE00000013). This manuscript reflects only the authors' views and opinions, neither the European Union nor the European Commission can be considered responsible for them.

ion rather than serially as RNNs require. This leads to better scalability since it is now possible to train efficiently on large datasets. However, Transformers cannot be implemented recursively and require to keep the entire sequence in memory for the attention operation with quadratic cost in the sequence length. A recent development in NLP, namely the RWKV model [7], has shown state-of-the-art results with a hybrid attention-recursion design, combining the power of an attention mechanism and the parallelization advantages of Transformers for training, as well as a low-memory recursive implementation for inference. We thus propose to utilize RWKV to design our model that processes one line at a time, where we can parallelize training over an entire hyperspectral image, but have low-memory recursive inference that only requires to store and process one line with all its bands (plus some internal states). The RWKV line operation is then coupled with additional neural network blocks, in a model we call LineRWKV, to predict pixel values based on a causal spatial and spectral context. Encoding prediction residuals with any entropy coding method completes our prediction-based compression model.

We report preliminary results on the recently introduced large-scale HySpecNet-11k dataset [8] showing substantial gains over the CCSDS-123 standard in lossless and near-lossless compression modes at an interesting, albeit unoptimized, complexity, confirming the promise of the approach.

2. BACKGROUND ON RWKV

Transformers [6] use the scaled dot-product attention mechanism to create highly powerful models for sequential data. A great advantage they have, which is pivotal to scaling, is that the attention operation can be easily parallelized. However, they suffer from high memory and computational complexity scaling quadratically with the sequence length. On the other hand, the older recurrent neural networks they supplanted offer linear scaling in memory and computations, at the price of operations not as effective as the attention mechanism and slow serial training. The Receptance Weighted Key Value (RWKV) model [7] combines the advantages of the two, with parallel Transformer-like training and a low-memory, linear-complexity recursive implementation for inference. Calling T the sequence length and d the feature dimensionality, a Transformer layer has $O(T^2d)$ time complexity (time refers to the sequence-length dimension) and $O(T^2 + Td)$ memory complexity, while RWKV has $O(Td)$ and $O(d)$ respectively. We refer the reader to [7] for a more complete overview of the architecture, but the fundamental operations performed by a RWKV layer on its input vector $\mathbf{x}_t$ are time-mixing:

$$\mathbf{r}_t = \mathbf{W}_r(\mu_r \mathbf{x}_t + (1-\mu_r)\mathbf{x}_{t-1}) \quad (1)$$

$$\mathbf{k}_t = \mathbf{W}_k(\mu_r \mathbf{x}_t + (1-\mu_r)\mathbf{x}_{t-1}) \quad (2)$$

$$\mathbf{v}_t = \mathbf{W}_v(\mu_r \mathbf{x}_t + (1-\mu_r)\mathbf{x}_{t-1}) \quad (3)$$

$$\mathbf{z}_t = \frac{\sum_{i=1}^{t-1} e^{-(t-1-i)w+\mathbf{k}_i}\mathbf{v}_i + e^{u+\mathbf{k}_t}\mathbf{v}_t}{\sum_{i=1}^{t-1} e^{-(t-1-i)w+\mathbf{k}_i} + e^{u+\mathbf{k}_t}} \quad (4)$$

$$\mathbf{o}_t = \mathbf{W}_o(\sigma(\mathbf{r}_t) \odot \mathbf{z}_t) \quad (5)$$

and channel-mixing:

$$\mathbf{o}_t = \sigma(\mathbf{r}_t) \odot (\mathbf{W}_v \max(\mathbf{k}_t, 0)^2) \quad (6)$$

where projection matrices $\mathbf{W}$ are learned during training.

It can be remarked that the core attention-like operation in Eq. (4) can also be written in a recursive manner, requiring only a state encoding the sum up to $t-1$ and the new input at time t. In this paper, we consider a sequence of lines with all the spectral bands, so that inference can work in a recursive manner in the satellite along-track direction.

3. METHOD

In this section we present the proposed method based on a neural network design we call LineRWKV. An overview is presented in Fig. 1. At a high level, we design a non-linear predictor, which, given a causal context of pixels, namely previous lines and bands, predicts pixels of a hyperspectral image so that only the prediction residual needs to be entropy encoded.

A core design principle of LineRWKV is that, during inference, the model only needs to store the current line to be encoded with all its spectral bands in order to limit memory consumption, which can be significant in spatio-spectral models of hyperspectral images. An efficient memory mechanism is therefore needed to keep track of past lines processed by the model. The traditional deep learning approach to this problem is through the use of recurrent neural network, such as LSTMs [9]. However, recurrent networks have recently fallen behind in favour of Transformers [6] due to the serial nature of their training process which does not allow scalability to large amounts of data. In our design we leverage the RWKV model, recently proposed in the language model literature, which is a hybrid design which admits a Transformer-like parallelization for training and a memory-efficient recurrent implementation for inference.

Referring to the schematic in Fig. 1, our proposed model is composed of the following main building blocks: i) a spatial encoder; ii) a line predictor; iii) a spectral predictor; iv) a pixel decoder;. Each of these building blocks can be implemented with neural networks. In particular, the *spatial encoder* is composed of a sequence of 1D convolution, LayerNorm, non-linearity operations with the goal of capturing the spatial context in line $y-1$ (i.e., the correlation across image columns) and encode it in a feature space. Its output is a vector of F features for each pixel of each band in the line. The *line predictor* uses the RWKV attention mechanism to predict the features of line y from the features of line $y-1$ produced by the encoder. A difference signal Δ in the feature

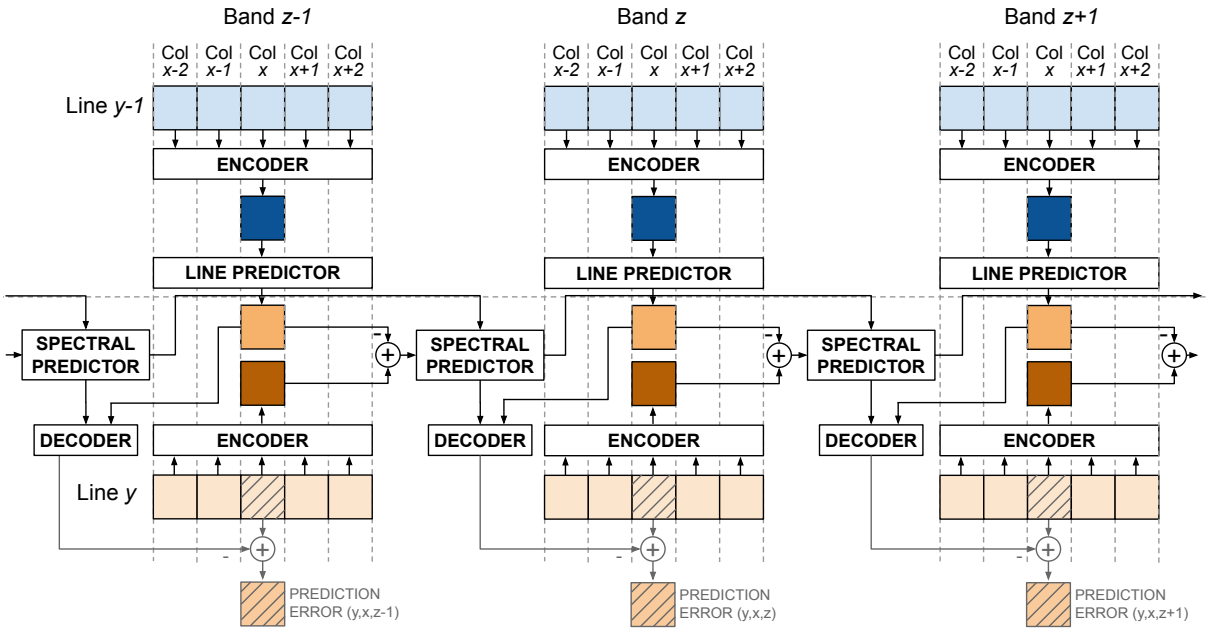

Fig. 1. Overview of LineRWKV. Encoder, decoder, line and spectral predictors are neural networks trained end-to-end to minimize the prediction error for all pixels.

space is computed between the predicted features and the encoding of line y. A *spectral predictor* uses the value of the feature difference from the past bands $z, z-1, ...$ to predict the features of the pixel at position $x, y, z+1$. In this paper, we use causal convolutional layers with a simple causal band attention mechanism for the spectral predictor, but a number of other designs, including RWKV are worth investigating in the future. The value of the pixel is obtained from the features of the spectral prediction as well as the spatial prediction with a *decoder* consisting of concatenation and 1×1 convolution. Overall, this mechanism is capable of capturing spatial and spectral dependencies with receptive fields and complexity that can be scaled according to one's needs. Notice the use of difference signals to let the networks work in a residual regime, hopefully to simplify the functions they approximate.

Also notice that the creation of a causal context requires the first line with all its bands and the first band for all the lines to be encoded with separate techniques. In particular, we encode the first line in the first band by DPCM over columns, and the first line in the other bands by spectral DPCM. All the other lines in the first band use a separate pixel decoder that decodes from Δ alone, i.e., only spatial prediction is performed.

The method is trained to minimize the ℓ_1 norm of the prediction residual for all pixels predicted by the neural network. During inference, the real-valued prediction is rounded to the nearest integer to compute integer residuals to be entropy-coded. This approach is different from the usual approach in deep learning lossless compression where a probability distribution over the discrete possible symbols is produced instead of a real-valued quantity. While this latter approach is potentially superior, as it allows end-to-end optimization of the correct criterion without the unmodeled inference rounding behavior, it is challenging for satellite images that have a large number of symbols due to their higher dynamic range (e.g., 16 bits instead of 8).

The method can also perform near-lossless compression in two different ways. The first way is by prequantization [10], i.e., the image to be compressed is quantized and then lossless compression is applied. This was studied to lead to insignificant sub-optimalities in the high-rate regimes [10]. Alternatively, in-loop quantization is possible, in which the reconstructed values are used for the predictions instead of the original. This requires some operations to be run serially, and possibly multiple times, leading to inefficient implementations, but is marginally superior in terms of rate-distortion performance. We believe prequantization is the preferred choice as it allows a more flexible design of the spectral prediction mechanism and faster implementation. Also notice that we only optimize for lossless compression, while never accounting for quantization during training. This could be suboptimal and deserves further investigation.

4. EXPERIMENTAL RESULTS

In this section, we present some experimental comparisons with the state-of-the-art approach to onboard compression, represented by the CCSDS-123 standard. The goal of our experiments is to determine if LineRWKV is a superior pre-

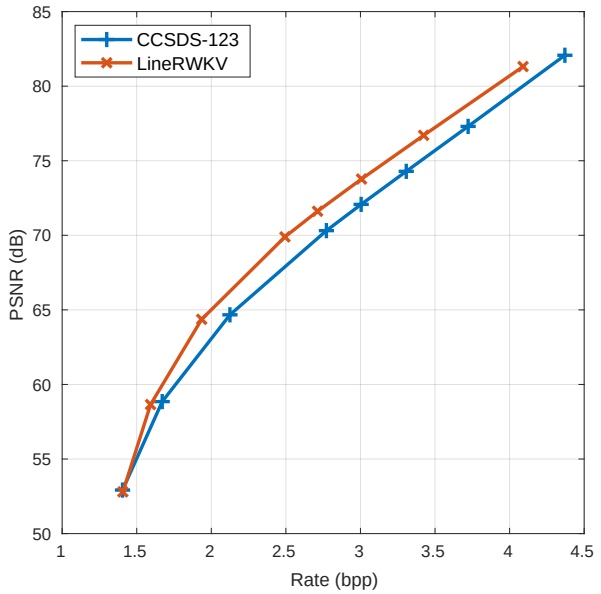

Fig. 2. Rate-distortion performance on HySpecNet-11k hard.

dictor to the CCSDS one at a reasonable complexity. In order to properly train and test the neural network, we used the recently proposed HySpecNet-11k dataset [8], which is the largest dataset of hyperspectral images currently available, composed of 11,483 non-overlapping patches of size $128 \times 128 \times 224$ acquired by the EnMAP satellite. We use the *hard* train-test split provided by the authors, where patches in the test set belong to entirely separate tiles to the training patches.

The LineRWKV model we use for the following experiments is composed of i) an encoder with 2 blocks of 3×1 convolution, LayerNorm, GeLU, producing $d = 64$ features; ii) a line predictor using 2 RWKV blocks with time-mixing, LayerNorm and channel-mixing; iii) a spectral predictor made of 2 layers of 3×1 causal convolution, LayerNorm, GeLU; iv) a decoder with one 1×1 convolutional layer. Overall, the model has roughly 110k trainable parameters. The model has been trained for 2000 epochs with a learning rate decayed linearly from 10^{-4} to 10^{-5} over four A100 40GB GPUs for two days. The data have been randomly subsampled to blocks of 16 contiguous bands to limit memory consumption, except for the last 50 epochs where 100 bands have been used. Notice that inference can use any number of bands but any difference with respect to what is used in training can lead to suboptimal results. Also notice that thanks to the 1×1 decoder, image columns can be subsampled after the encoding process, which can be useful to limit memory usage during training with a large number of bands. We remark that these high memory requirements are only for training in order to ideally process the full image size in a parallel Transformer-like fashion. The inference process has, instead, modest memory requirements, compatible with onboard usage.

Fig. 2 shows the rate distortion curve achieved by LineRWKV when compared with CCSDS-123. The result is averaged over the entire hard test set. LineRWKV uses prequantization, while CCSDS uses in-loop quantization. CCSDS is using the "full, neighbor-oriented" mode which is known to typically provide the best rate-distortion performance, with 3 prediction bands. Both methods use the Golomb entropy coder specified by the standard for a fair comparison. The lossless rate achieved by CCSDS is 5.801 bpp, compared to 5.593 bpp achieved by LineRWKV, thus achieving a significant reduction of 0.208 bpp. This confirms that LineRWKV is a superior predictor compared to the adaptive filter used in the CCSDS standard.

Concerning complexity, the proposed model is estimated to require around 120k FLOPs/sample. We benchmarked it on a desktop GPU using only single-precision floating point FP32 operations to have an encoding throughput of 1.86 Msamples/sec. Future work will optimize it and test on a low-power neural network accelerator, but we conjecture that a throughput of 1 Msample/sec on a 5 W chip is achievable. Peak memory usage is benchmarked to be around 1200 MB, which a rather low value concerning processing hyperspectral images. This is due to the efficient line-by-line processing of the architecture. Significant optimizations are possible in both throughput and memory usage by considering neural network quantization, mixed-precision approaches and sparsification.

5. CONCLUSIONS

We presented a novel neural network design for onboard predictive coding of hyperspectral images. Its line-based operation, coupled with the recurrent-attentive mechanism of RWKV enables to combine the efficient inference of recurrent networks and the power of Transformers. Preliminary results show significant performance gains in lossless and near-lossless compression against CCSDS-123 and a promising throughput.

6. REFERENCES

[1] Consultative Committee for Space Data Systems (CCSDS), "Low-Complexity Lossless and Near-Lossless Multispectral and Hyperspectral Image Compression," *Blue Book*, , no. 1, February 2019.

[2] Johannes Ballé, Valero Laparra, and Eero P Simoncelli, "End-to-end optimized image compression," *arXiv preprint arXiv:1611.01704*, 2016.

[3] Yaman Dua, Ravi Shankar Singh, Kshitij Parwani, Smit Lunagariya, and Vinod Kumar, "Convolution neural network based lossy compression of hyperspectral images," *Signal Processing: Image Communication*, vol. 95, pp. 116255, 2021.

[4] Riccardo La Grassa, Cristina Re, Gabriele Cremonese, and Ignazio Gallo, "Hyperspectral data compression using fully convolutional autoencoder," *Remote Sensing*, vol. 14, no. 10, pp. 2472, 2022.

[5] Sebastià Mijares i Verdú, Johannes Ballé, Valero Laparra, Joan Bartrina-Rapesta, Miguel Hernández-Cabronero, and Joan Serra-Sagristà, "A scalable reduced-complexity compression of hyperspectral remote sensing images using deep learning," *Remote Sensing*, vol. 15, no. 18, 2023.

[6] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[7] Bo Peng, Eric Alcaide, Quentin Anthony, Alon Albalak, Samuel Arcadinho, Huanqi Cao, Xin Cheng, Michael Chung, Matteo Grella, Kranthi Kiran GV, et al., "RWKV: Reinventing RNNs for the Transformer Era," *arXiv preprint arXiv:2305.13048*, 2023.

[8] Martin Hermann Paul Fuchs and Begüm Demir, "Hyspecnet-11k: A large-scale hyperspectral dataset for benchmarking learning-based hyperspectral image compression methods," *arXiv preprint arXiv:2306.00385*, 2023.

[9] Sepp Hochreiter and Jürgen Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[10] Diego Valsesia and Enrico Magli, "High-throughput onboard hyperspectral image compression with ground-based cnn reconstruction," *IEEE transactions on geoscience and remote sensing*, vol. 57, no. 12, pp. 9544–9553, 2019.