QUASI LOSSLESS SATELLITE IMAGE COMPRESSION

Pascal Bacchus[1]*, Renaud Fraisse*[2]*, Aline Roumy*[1]*, Christine Guillemot*[1]

[1]INRIA, Rennes, France, [2]Airbus Defence and Space, Toulouse, France
[1] *name.surname*@inria.fr, [2] renaud.fraisse@airbus.com

ABSTRACT

We describe an end-to-end trainable neural network for satellite image compression. The proposed approach builds upon an image compression scheme based on variational autoencoders with a learned hyper-prior that captures dependencies in the latent space for entropy coding. We explore this architecture in light of specificities of satellite imaging: processing constraints onboard the satellite (complexity and memory constraints) and quality needed in terms of reconstruction for the processing task on the ground. We explore data augmentation to improve the reconstruction of challenging image patterns. The proposed model outperforms the current standard of lossy image compression onboard satellite based on JPEG 2000, as well as the initial hyper-prior architecture designed for natural images.

***Index Terms*—** Deep Image Compression, Neural Networks, Satellite Application.

1. INTRODUCTION

The last years have seen an explosion in earth observation data of increased resolution, with obvious implications on the volume of data to be processed onboard the satellite, which can exceed billions of pixels per image. The transmission to the ground of this large volume of data requires efficient compression solutions which can preserve the high frequency details needed for interpretation tasks to be performed on the ground. The solution currently used has been defined by the consultative committee for space data systems [1] and is based on JPEG 2000. This solution uses a handcrafted DCT transform followed by quantization and entropy coding.

Satellite image compression algorithms must fulfil three characteristics. First, (i) quasi-lossless compression is needed in order to allow accurate on ground interpretation. Second, (ii) the algorithms must be of low computational complexity. Third, (iii) satellite images differ from natural images (images captured from a handheld photographic camera), as they contain small objects (as small as the size of the pixels) and have high entropy, and thus compression must preserve these high frequency details. While the literature on satellite image compression mostly focuses on multi and hyper-spectral images [2, 3], we propose a single image compression algorithm based on variational auto-encoders (AEs), that preserves the three required properties.

The field of image compression has indeed recently known significant advances based on neural networks. Neural networks based solutions outperform traditional codecs [4, 5] both in terms of visual quality and quantitative measures, i.e., perceptual (SSIM) and distortion (PSNR) metrics.

This progress has been made possible thanks to the use of variational AEs [6, 7] that are learned end-to-end to compress the input data into a lower dimensional latent space. The latent representation is quantized and entropy coded. the authors in [8, 9] further propose a so-called hyper-prior based on an AE that learns the parameters of the statistical distribution of the latent representation for entropy coding. Note that the use of variable AEs for satellite image compression has already been explored in [10, 11] where the authors propose a simplified entropy model to reduce the complexity of the AE based hyper-prior of [8].

In this paper, we further explore the use of variational auto-encoders for satellite image compression to reach all goals (i), (ii) and (iii). In order to achieve an accurate rate-distortion trade-off, an attention module is trained to allow more rate to encode the (true) high frequency details of the image (goal (i) and (iii)). Moreover, to preserve pixel-size details (iii) in all directions, we augment the training data and transform them with a shear mapping. Finally, small filters size and light normalization function are considered to reduce the computational complexity (goal (ii)).

2. VARIATIONAL AUTO-ENCODER BASED IMAGE COMPRESSION: BACKGROUND

The hyper-prior architecture [8] is made of two AE networks as shown in Figure 1. The first AE receives the original image x and generates a latent representation y. Quantization and entropy coding is performed to produce $\hat{y}$ which is decoded by the inverse transform to reconstruct $\hat{x}$. The purpose of the other AE (the hyper-prior) is to extract the parameters of the latent representation distribution to enhance the entropy model. This entropy model is shared between the encoder and decoder and used to code the quantized latent representation into a bit-stream. This allows having entropy coding models adapted to the characteristics of a specific image as the

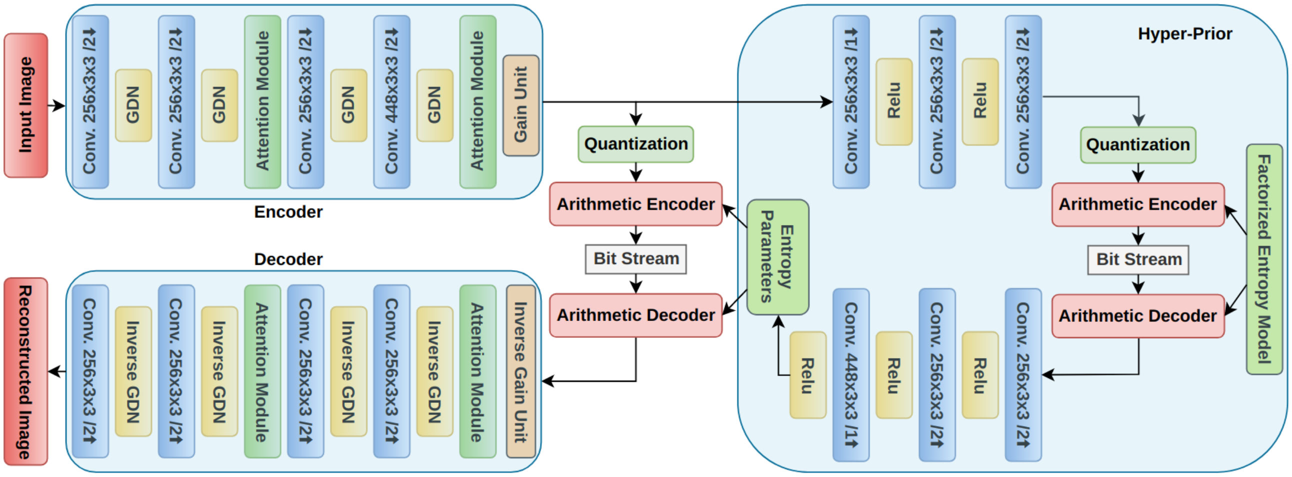


Fig. 1. The architecture of our end-to-end compression model.

entropy parameters are estimated for each image.

The parameters are learned with the following optimization problem: a trade-off between a distortion $D(x, \hat{x})$ between the original image x and the reconstructed image $\hat{x}$ and the rate $R(\hat{y})$ of the generated bit-stream.

$$J = \lambda D(x, \hat{x}) + R(\hat{y}) \tag{1}$$

where λ is a factor to balance each term of the equation (1), and the distortion is chosen as MSE. This equation which represents our loss function is then minimized through backpropagation. In the context of compression, the derivative of the quantization function is either zero or undefined. To overcome the issue of non-differentiability, the quantization is replaced by uniform noise during training.

Generalised Divisive Normalization (GDN) has been shown [6] to be highly efficient in shaping the local joint statistics of images into Gaussians. The GDN layer is used in its simplified form [12]:

$$z_i = \frac{x_i}{\beta_i + \sum_j \gamma_{ij} x_j} \tag{2}$$

where z is the filter response and γ, β are learned parameters. It suffers from a minor drop in performance for sensible gain in computational complexity.

3. PROPOSED ARCHITECTURE

To address the specific needs of satellite imaging, we choose to work with the standard version of the hyper-prior network [8] instead of more advanced and effective networks [9, 4, 5] as they are less suited to the efficient use of GPUs. We modify parameters of the hyper-prior model and augment it with gain units [13], attention modules [14] and shear mapping to respond to low computational complexity and high reconstruction quality. The whole network architecture is detailed in Figure 1.

3.1. Reduce inference time

Models targeting high bit rates can suffer from saturation in their performance gains if their capacity (i.e. model complexity in the number of nodes and parameters) is not high enough as shown in the appendix of [8]. The number of filters is thus increased from 192 (analysis transform) and 256 (bottleneck part) to respectively 256 and 448. The size of filters is reduced from 5 to 3 to reduce complexity and also because it has no drop in distortion performance. Satellite images have more information per pixel than natural images due to their higher entropy, so the filter size does not need to be large to capture the information in the area of a pixel.

Most end-to-end AEs compression models [8, 9, 4, 5] are trained for a specific rate-distortion point, this requires training and loading several models onboard and waste storage during run time. To reply to this efficiency demand, gain units are added at the end and beginning of respectively the encoder and decoder as done in [13] with the hyper-prior compression network. We simplify it by not distinguishing between the feature maps and applying the same scaling everywhere. It allows for variable bit rate for a single model and works as a quality parameter added during inference time while reducing the memory consumption.

3.2. Enhance the reconstruction quality

To further increase compression and better adapt to pixel-sized details, data augmentation is performed before each batch of images. The issue is not the total number of images we trained our network on, as the overall compression generates good results. But, some small patterns are badly compressed (e.g. striped patterns) and yield a high reconstruction error. Data augmentation through the use of shear mapping (mapping based om shearing transform) paired with rotation is aimed at those challenging structures and increase the number of occurrences during training. It also helps to reduce overfitting and thus acts as a regularizer.

Finally, the use of attention modules [14] is motivated by the performance obtained in computer vision tasks. In the context of image classification, those layers are used to discard non-relevant background information. The learning of this trade-off for each data set depends on the context and is driven by gradient descent. On the subject of image compression, this layer helps the network highlight the challenging part of the image to balance the bit rate between edges, high frequencies and texture. We are using a lightweight version [5] that comes with a significant reconstruction gain for low computational complexity added.

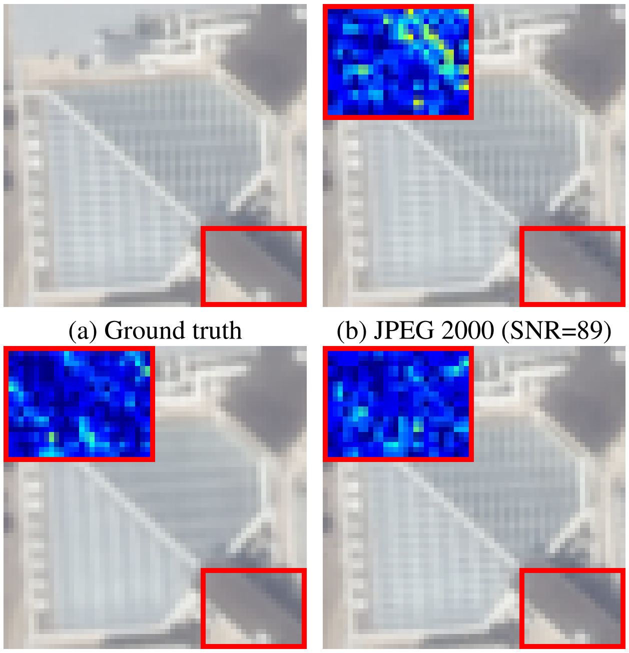
4. EXPERIMENTS

4.1. Training details

All code is using parts of CompressAI [15] a PyTorch library for deep learned compression model. The data set we used is made of 300 12-bits RGB satellite images (2000x2000), with a geometric resolution (effective ground distance between two pixels) of 50cm, that are then cropped to form 4800 patches (500x500). 5% are used for testing, the rest for training. We use an initial learning rate of $1e^{-4}$ which is halved when the evaluation loss reaches a plateau of 10 epochs. The lambda value used to balance the rate-distortion trade-off is 1.25 to target a medium bit rate (around 2 bpp). Data augmentation is randomly performed (on average 50%) on each batch with rotation and shear mapping. Experiments are conducted on an NVIDIA QUADRO RTX 8000 GPU. Training time is around 5 hours for 200 epochs and inference time (without model loading) is about 1s to encode a 2000x2000 image and 1.5s to decode it.

4.2. Qualitative results

We visually compare in Figure 2, at the same bit rate, our model with the ground truth and JPEG 2000. The differences between all images are slight as we compare high bit-rate images, a requirement to obtain the most of high geometric resolution images. The influence of shear mapping in data augmentation is clear in striped patterns that tend to be blurry in the deep learned model even though the overall compression achieves a greater SNR. This is illustrated in Figure 3, with the Fourier transform graph. Shear mapping allows the network to explore a larger part of the spectrum and keep more very high frequencies information which is the case for 1-pixel stripped patterns. Without this type of data augmentation, the network still has a good reconstruction overall but act as a low-band filter. When zooming into fine details, the JPEG2000 compressed image suffers from colour artefacts, especially in the uniform textured pattern. However, both learned models are close to one another and to the ground truth image.


(a) Ground truth (b) JPEG 2000 (SNR=89)

(c) No shearing (SNR=119) (d) Our model (SNR=111)

Fig. 2. Visual comparison of compressed images (2 bpp) with the ground truth. Geometric resolution of 50cm. (c) is our model without shear mapping during training. The enlarged image corresponds to the error map of the area compared to the ground truth (range [0;6]).

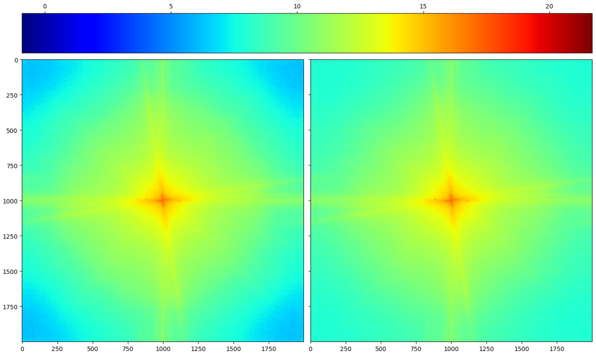


Fig. 3. Left: Our model without shear mapping, Right: Our model

4.3. Quantitative results

We evaluate our model on a subset of representative satellite images. We are using SNR as our preferred metric as it allows us to easily compare the amount of noise added by the compression with noise already present in the image. We compare our model with JPEG 2000, which has been used to define the standard for onboard satellite compression [1]. Note that [1] and JPEG 2000 have very close performance as shown in [10]. A perk of neural network models is that they are data-driven algorithms thus they can extract all the information that defines a particular scenery. The hyper-prior model has been designed for natural images but a training including only satellite images yields a great improvement as the network's weights adapt to this kind of image as seen in Figure 4. To measure the average gain in SNR or bit-rate between two rate-distortion curves, we use the Bjontegaard metric [16]. The difference between the baseline model trained on natural images and the one trained on satellite data is: BD-SNR : 22.2 SNR, BD-RATE: -21.7%. Thus training with satellite images alone reduces the bit rate by 21%.

Shear mapping models are significantly lower in performance than even the standard hyper-prior model. The care on challenging patterns provided during training is done at the expense of a good reconstruction metric. The opposite can be seen with an attention only model where the overall SNR is greater but some artefacts remain in striped patterns. Also, it quickly saturates at a high bit rate compared to other deep models. Our model which includes both attention modules and shear mapping, can mitigate the shear mapping downside and ensure a better reconstruction on average than the hyper-prior model while being able to preserve high frequency details. Our model surpasses the hyper-prior model by BD-SNR: 5.7 SNR, BD-RATE: -5.4%.

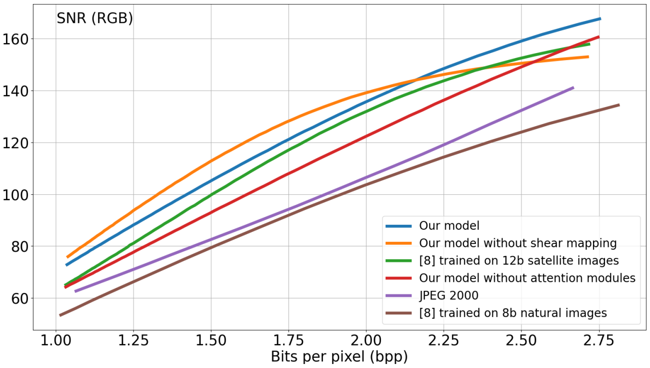


Fig. 4. Our combined approach has comparable performance to the attention model but over a wider range of bit rates while retaining more high frequency detail.

5. CONCLUSION

In this work, we have proposed a deep learned framework with improved distortion performance compared to the current baseline but without sacrificing its perceptual quality with challenging patterns. Even high frequencies are well reconstructed to keep as many details as possible at a pixel level. The inference time and the network variable rate capability makes it well suited to onboard constraints. It gives a promising step for extending it to higher resolution images and includes more image processing onboard satellites such as demosaicking to work with raw data.

6. REFERENCES

[1] Consultative Committee for Space Data Systems (CCSDS), *Image data compression CCSDS 122.0-B-1*, CCSDS, 2005.

[2] G. Yu, T. Vladimirova, and M. N. Sweeting, "Image compression systems on board satellites," *Acta Astronautica*, vol. 64, no. 9, pp. 988–1005, 2009.

[3] Z. Wang, B. Gao, P. Wang, X. Gong, and L. Tong, "High-quality fast compression algorithm based on fractal-wavelet," in *IGARSS*, 2021, pp. 3900–3903.

[4] D. Minnen and S. Singh, "Channel-wise autoregressive entropy models for learned image compression," in *ICIP*, 2020.

[5] Z. Cheng, H. Sun, M. Takeuchi, and J. Katto, "Learned image compression with discretized gaussian mixture likelihoods and attention modules," in *CVPR*, 2020.

[6] J. Ballé, V. Laparra, and E. P. Simoncelli, "End-to-end optimized image compression," in *ICLR*, 2017.

[7] L. Theis, W. Shi, A. Cunningham, and F. Huszár, "Lossy image compression with compressive autoencoders," in *ICLR*, 2017.

[8] J. Ballé, D. Minnen, S. Singh, S. J. Hwang, and N. Johnston, "Variational image compression with a scale hyperprior," in *ICLR*, 2018.

[9] D. Minnen, J. Ballé, and G. D. Toderici, "Joint autoregressive and hierarchical priors for learned image compression," in *NeurIPS*, 2018.

[10] V. Alves de Oliveira, M. Chabert, T. Oberlin, C. Poulliat, M. Bruno, C. Latry, M. Carlavan, S. Henrot, F. Falzon, and R. Camarero, "Reduced-complexity end-to-end variational autoencoder for on board satellite image compression," *Remote Sensing*, vol. 13, no. 3, 2021.

[11] V. Alves de Oliveira, M. Chabert, T. Oberlin, C. Poulliat, M. Bruno, C. Latry, M. Carlavan, S. Henrot, F. Falzon, and R. Camarero, "Satellite image compression and denoising with neural networks," *IEEE Geoscience and Remote Sensing Letters*, vol. 19, 2022.

[12] N. Johnston, E. Eban, A. Gordon, and J. Ballé, "Computationally efficient neural image compression," *arXiv preprint arXiv:1912.08771*, 2019.

[13] T. Chen and Z. Ma, "Variable bitrate image compression with quality scaling factors," in *ICASSP*, 2020.

[14] S. Woo, J. Park, J. Lee, and I. S. Kweon, "Cbam: Convolutional block attention module," in *ECCV*, 2018.

[15] J. Bégaint, F. Racapé, S. Feltman, and A. Pushparaja, "Compressai: a pytorch library and evaluation platform for end-to-end compression research," *arXiv preprint arXiv:2011.03029*, 2020.

[16] G. Bjøntegaard, "Calculation of average psnr differences between rd-curves," 2001, VCEG-M33.