Encoding hyperspectral data with low-bond dimension quantum tensor networks

Fabian Fischbach[1,†], Hans-Martin Rieser[1,*] and Oliver Sefrin[2,†]

1- DLR Institute for AI Safety and Security - Dept. for Execution Environments - Sankt Augustin & Ulm - Germany

2- DLR Institute of Quantum Technologies - Dept. for Quantum Information and Communication - Ulm - Germany

[†]First authors [*]Second authors

Abstract. Encoding data on a quantum computer poses a major challenge on data intensive quantum applications like machine learning. In particular, data with complex internal structure like emission spectra need to be adapted to reduce the encoding effort of quantum circuits. We empirically investigate the influence of compression on the encoding of hyperspectral data into quantum states, to make its encoding more efficient. To this end, we assess the effect of approximating states by low-bond dimension matrix product states fed into a variational quantum classifier on the public Pavia University benchmark dataset.

1 Introduction

Machine learning (ML) has grown an energy intensive business, both for training and inference. The rise of quantum computers (QCs) with computational possibilities beyond classical hardware may yield improvements also for ML purposes. However, quantum physics prohibits copying and re-using quantum states. Therefore, data needs to be loaded in each iteration of the quantum model, requiring efficient data encoding methods for making QCs useful in ML.

For images, several promising encoding ansätze have been developed. In amplitude encoding (AE), the classical values of the pixels are mapped to the coefficients of a quantum basis, which is the most space efficient method at the cost of the circuit depth required for creating the state. The Flexible Representation of Quantum Images (FRQI) [1] separates the description in value qubits and address qubits. It can be generalized to RGB images (Multi-Channel representation for Quantum Images, MCQI) with polynomial gate complexity [2].

In this work, we expand the existing description to hyperspectral (HS) images with a high number of bands and compress the state vector using tensor network (TN) techniques before encoding on a QC, similar to refs. [3, 4, 5]. This approximation greatly reduces the circuit depth needed for mapping the information subsequently to the QC. We focus on assessing whether this compression of HS images affects classification results. This is particularly important for this type of image due to the high amount of information per pixel, containing redundancies from a classification standpoint.

2 Tensor networks

TNs are mathematical objects for storing and manipulating complex tensorial structures in a memory efficient form, originally developed for numerical simulations of large quantum systems [6]. TNs also proved useful for ML and data compression, especially in combination with QCs, as their inherent structure makes mapping between classical and quantum devices straightforward [7].

Graphically, TNs consist of low-dimensional nodes, representing the degrees of freedom, and edges (bonds), representing tensor contractions. When applied, the contractions yield a single tensor, which contains the same information, but typically has redundancies that raise storage and handling effort. Inversely, a TN can be derived from any tensor by decomposition methods like singular value decomposition (SVD). SVD cuts part of the tensor into a new tensor node that is connected to the remainder tensor via an internal bond of a size proportional to the information shared by both subsystems. This size is called bond dimension χ.

The most common TN layout is a matrix product state (MPS) of the form $A_a^{\sigma_1} B_b^{a\sigma_2} C_c^{b\sigma_3}$..., which we also use here. A major advantage of an MPS is its relatively simple mapping to QCs: first, one brings the MPS to an isometric canonical form by iteratively applying contractions and SVD on the tensor nodes. Second, the isometries are promoted to unitary tensors by padding them with a basis for their kernel. Each unitary corresponds to a quantum gate where the padded dimensions correspond to ingoing ancilla qubits or outflowing discarded qubits [7]. The MPS is mapped to a staircase shaped array of gates of sizes depending on χ (see Fig. 2). Lossy compression is realized by truncating during SVD and therefore limiting the maximal χ allowed. This also limits the number of in- and outgoing qubits per logical gate and thus yields shorter circuits when compiled to hardware-native gates.

3 Study setup

3.1 Dataset

We use the Pavia University (PU) dataset acquired by the airborne imaging spectrometer ROSIS over an urban area surrounding the University of Pavia (Italy). It has 103 spectral bands of size 610×340 pixels, a spectral coverage from 430 nm to 860 nm, and a spatial resolution of 1.3 m. Fig. 3 shows the ground truth, involving nine labeled classes. Ref. [8] provides a standardized dataset version (henceforth PU-DASE) with only a subset of 3921 pixel labels retained for users to train their models. A second subset of 10 759 pixel labels is held back for evaluating user uploaded classification maps in terms of accuracy metrics. Our metrics of interest for this problem are the *overall accuracy* (OA), which is the total ratio of correctly classified samples to all samples, and the *average accuracy* (AA), which is the mean of the accuracy ratios for each label class.

3.2 Encoding schemes

MCQI Let $I_{ij}^c \in [0,1]$ be the intensity of color channel $c \in \{0, 1, \ldots, C-1\}$ of the pixel at position (i,j) and set $\theta_{ij}^c = \arccos I_{ij}^c$. We pad the channel

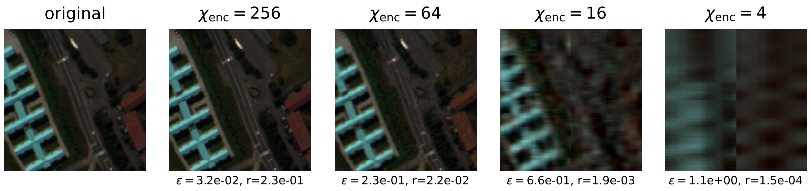

original
$\chi_{\text{enc}} = 256$
$\chi_{\text{enc}} = 64$
$\chi_{\text{enc}} = 16$
$\chi_{\text{enc}} = 4$
ε = 3.2e-02, r=2.3e-01
ε = 2.3e-01, r=2.2e-02
ε = 6.6e-01, r=1.9e-03
ε = 1.1e+00, r=1.5e-04

Fig. 1: Compression error and rate for 128 × 128 patches (false-color image).

vectors to $\tilde{C} = 2^{\lceil \log_2(C) \rceil}$ channels and fix $\theta^c_{ij} = 0$ for all $c \geq C$. An ordering of the n_{pix} pixels (e.g. snake-like) is given by a sequence of integers $n_{i,j} \leftrightarrow (i,j)$ $(0 \leq n_{i,j} < n_{\text{pix}})$. The image maps to $1 + \lceil \log_2(C) \rceil + \lceil \log_2(n_{\text{pix}}) \rceil$ qubits via:

$$|\rho_{ij}\rangle_{\text{col}} = \sum_{c=0}^{\tilde{C}-1} \left(\cos\theta^c_{ij} |0\rangle + \sin\theta^c_{ij} |1\rangle \right) |c\rangle \ , \quad |\psi\rangle = \frac{1}{\mathcal{N}} \sum_{i,j} |\rho_{ij}\rangle_{\text{col}} |n_{i,j}\rangle \ . \tag{1}$$

Here $\{|c\rangle\}$ denotes the computational basis on the color qubit space and $\{|n_{i,j}\rangle\}$ for the address qubits. $\mathcal{N}$ is a normalization constant. The computational complexity of preparing an MCQI state $|\psi\rangle$ is the same as for FRQI, i.e., $O(n^2_{\text{pix}})$ [2].

AE A spectral vector $(I^c_{ij})_{c=0,\ldots,\tilde{C}-1}$ is divided by its Euclidean norm directly yielding the real coefficients of a superposition of $\tilde{C}$ computational basis states. A disadvantage is the loss of the overall brightness of a pixel. By replacing $|\rho_{ij}\rangle_{\text{col}}$ with an AE spectrum in eq. (1), one can use an addressing scheme to encode image patches instead of single pixels[1].

Compression The resulting state vector is a tensor of $2^{n_{\text{qubits}}}$ complex entries. Using sequential SVD as described in section 2, this tensor is converted to an MPS of limited $\chi \leq \chi_{\text{enc}}$. Fig. 1 visualizes the effect of encoding an image patch with MCQI and compressing it with varying limit χ_{enc}. By denoting the tensor holding all coefficients of the MCQI-encoded state as I_{orig}, we quantify the *compression error* ϵ by the normalized Frobenius norm between I_{orig} and the MPS-compressed and re-contracted tensor I_{compr}, $\epsilon = \frac{\|I_{\text{orig}} - I_{\text{compr}}\|_F}{N}$, where N is the number of tensor elements. With the *compression rate* r we denote the ratio of tensor elements between the MPS-compressed and the MCQI tensor.

3.3 Classification

We consider the impact of MPS compression on multi-class classification using individual pixel data as input. Fig. 2 schematically shows our quantum circuit with data encoding (blue) and a variational quantum classifier (VQC) (red/orange): First, a state vector of AE or MCQI type is prepared for each input as a single MPS layer of limited bond dimensions $\chi \leq \chi_{\text{enc}} = 2^2$. This requires seven qubits for amplitude and eight qubits for MCQI encoding. The encoding part is followed by $n = 3$ MPS layers of parametrized two-qubit SU(4)-gates (15 parameters per gate). Pauli-z measurements on the last m qubits provide a probability distribution for 2^m bit strings, interpreted as up to 2^m possible class labels. The

[1] The (i,j)-sum must then skip potential null spectra that occur for zero-padded images.

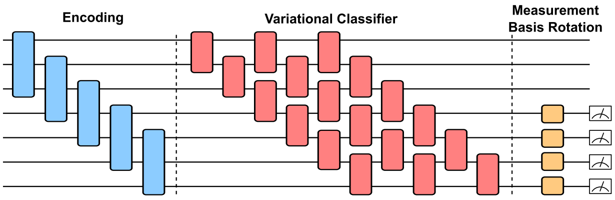


Fig. 2: An example of a quantum circuit for multi-class classification (with $\chi_{\text{enc}} = 2^2$, seven qubits for amplitude encoding, and three MPS layers).

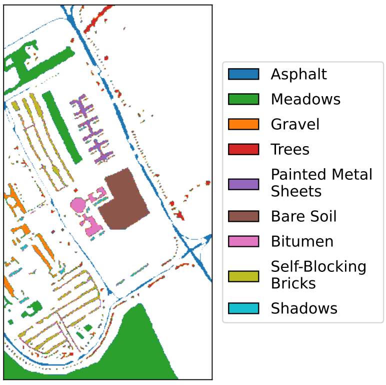


Fig. 3: PU ground truth.

Parameter	Value
Epochs	5000
Batch Size	120
Loss	`CategoricalFocal-Crossentropy` ($\alpha = 0.25$, $\gamma = 1.0$)
Optimizer	`Adam` ($\beta_1 = 0.9$, $\beta_2 = 0.999$)
Initial LR	5×10^{-4}
Weights Initial.	Random uniform with interval $[-\pi, \pi]$

Table 1: Training hyperparameters.

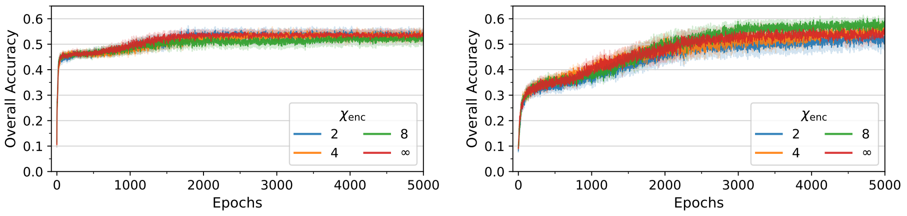
PU dataset requires $m = 4$. Motivated by [5], a final layer of single-qubit gates (orange) allows for local basis transformations before the measurement, leading to 282 (327) trainable weights in total for AE (MCQI) encoding.

All noise-free state vector simulations use PennyLane version 0.36.0 [9], gradient training uses TensorFlow (TF) version 2.15.0 [10] on 80 % of the PU-DASE set, while the remaining 20 % comprise the validation dataset. The hyperparameters, loss function and optimizer are given in Tab. 1. The TF callback `ReduceLROnPlateau`[2] was found to improve the overall accuracies and thus chosen for the main experiment. After training, we use the model weights from the epoch with the best overall accuracy on the validation subset to generate the prediction map on the PU-DASE test subset. As the main experiment is to probe the influence of the MPS compression on the classifiability of the input quantum state, we train both with MCQI and AE encoding and different bond dimension limits $\chi_{\text{enc}} \in \{2, 4, 8, \infty\}$ using 10 different random initializations. Training times are $\mathcal{O}(100)$ 8-core-hours per parameter configuration and seed.

4 Results

Fig. 4 shows the training progress for the different bond dimension limits for AE (4a) and MCQI (4b). For both encoding types, no significant effect with respect to χ_{enc} can be discerned. Regarding the encoding, we note that the

[2] Monitoring validation loss, operating w. reduction factor 0.5, patience 100 and Δ_{min}=10^{-4}.



(a) Amplitude Encoding

(b) MCQI Encoding

Fig. 4: OA on the training set (mean and standard error over 10 random seeds).

training runs using AE converge much faster after roughly 2000 episodes, whereas MCQI runs appear to converge just towards the end of training.

Tab. 2 shows the overall and average accuracy for each configuration. We have aggregated the ten prediction maps as an ensemble using a majority vote before evaluating on the PU-DASE test subset. Again, we cannot observe a clear deterioration trend of the accuracy towards smaller bond dimension limits (higher compression). With respect to the encoding, however, classifiers using AE achieve between roughly 5 to 8 percent points higher OA and 8 to 13 percent points higher AA than their MCQI counterparts. Lower OAs for MCQI on the PU-DASE test subset compared to the final OAs on the training subset (Fig. 4b) indicate weaker generalization capability than AE.

The confusion matrix for the ensemble prediction with the highest OA using AE and $\chi_{\text{enc}} = \infty$ is presented in Fig. 5. Of the nine classes, two are predicted with perfect accuracy, three more have an accuracy greater than 80 %, and two reach at least 50 %. *Meadows* (3 %) and *Gravel* (0 %), however, are barely learned. *Meadows* samples are largely classified as *Trees* and for *Gravel* not a single sample is predicted into said class. The reason for that is the high similarity to the spectra of other classes that is not captured by the setup.

Encoding	χ_{enc}	OA [%]	AA [%]
MCQI	2	46.1	55.1
	4	44.8	53.2
	8	45.2	54.7
	∞	46.8	54.4
AE	2	50.7	62.7
	4	52.2	66.5
	8	53.5	65.4
	∞	53.9	64.5

Table 2: Results aggregated over ten random seeds per configuration (majority vote).

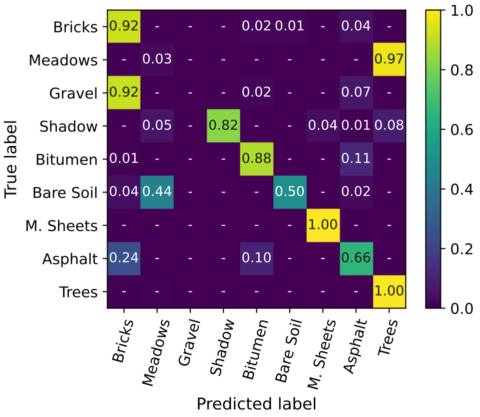


Fig. 5: Confusion matrix for the ensemble prediction with amplitude encoding and $\chi_{\text{enc}} = \infty$.

5 Conclusion

In this work, we have probed the compressibility of hyperspectral image data using quantum tensor networks, in particular for quantum machine learning. Our numerical experiments on the PU benchmark data set have shown for both the amplitude and the MCQI representation that even strong MPS compression, i.e., choosing smallest bond dimension limits, does not significantly deteriorate the single-pixel accuracies of a variational quantum classifier. Former encoding typically leads to better overall accuracies. Our main interpretation is, that the raw spectra contain redundant information that can be efficiently compressed with TN methods from the viewpoint of our specific classifier. A classifier with higher expressivity will probably require larger maximal bond dimensions.

Natural further research tasks are to verify the findings on other benchmark sets, with alternative VQC ansätze and to include proximity information in inputs. Also, other TN layouts or variational encodings are worth consideration. However, reaching for VQC accuracies comparable to literature values for purely classical methods [11] remains a big challenge, not least due to prohibitive (simulation) resource demands when scaling up the model complexity (expressivity). The same holds for the inclusion of circuit noise.

References

[1] P. Q. Le, F. Dong, and K. Hirota, “A flexible representation of quantum images for polynomial preparation, image compression, and processing operations,” *Quantum Inf. Process.*, vol. 10, no. 1, pp. 63–84, 2011.

[2] B. Sun *et al.*, “An RGB multi-channel representation for images on quantum computers,” *J. Adv. Comput. Intell. Intell. Inform.*, vol. 17, no. 3, pp. 404–417, 2013.

[3] R. Dilip *et al.*, “Data compression for quantum machine learning,” *Phys. Rev. Res.*, vol. 4, p. 043007, 2022.

[4] B. Jobst *et al.*, “Efficient MPS representations and quantum circuits from the Fourier modes of classical image data,” 2023.

[5] K. Shen *et al.*, “Classification of the Fashion-MNIST Dataset on a Quantum Computer,” 2024.

[6] S. R. White, “Density matrix formulation for quantum renormalization groups,” *Phys. Rev. Lett.*, vol. 69, pp. 2863–2866, Nov 1992.

[7] H.-M. Rieser, F. Köster, and A. P. Raulf, “Tensor networks for quantum machine learning,” *Proc. R. Soc. A.*, vol. 479, no. 2275, p. 20230218, 2023.

[8] F. Dell’Acqua *et al.*, “The IEEE GRSS data and algorithm standard evaluation (DASE) website: Incrementally building a standardized assessment for algorithm performance,” in *2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pp. 2601–2608, 2017.

[9] V. Bergholm *et al.*, “Pennylane: Automatic differentiation of hybrid quantum-classical computations,” 2022.

[10] M. Abadi *et al.*, “TensorFlow: A system for Large-Scale machine learning,” in *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)*, (Savannah, GA), pp. 265–283, USENIX Association, 2016.

[11] N. Audebert, B. Le Saux, and S. Lefevre, “Deep learning for classification of hyperspectral data: A comparative review,” *IEEE Geoscience and Remote Sensing Magazine*, vol. 7, no. 2, pp. 159–173, 2019.