Triplet Loss Based Quantum Encoding for Class Separability

Marco Mordacci[1,*], Mahul Pandey[1,*], Paolo Santini[1] and Michele Amoretti[1]

[1]*University of Parma, Parco Area delle Scienze, Parma, 43124, Italy*

Abstract

An efficient and data-driven encoding scheme is proposed to enhance the performance of variational quantum classifiers. This encoding is specially designed for complex datasets like images and seeks to help the classification task by producing input states that form well-separated clusters in the Hilbert space according to their classification labels. The encoding circuit is trained using a triplet loss function inspired by classical facial recognition algorithms, and class separability is measured via average trace distances between the encoded density matrices. Benchmark tests performed on various binary classification tasks on MNIST and MedMNIST datasets demonstrate considerable improvement over amplitude encoding with the same VQC structure while requiring a much lower circuit depth.

Keywords

Quantum Machine Learning, Quantum Encoding, Triplet Loss

1. Introduction

The goal of quantum machine learning (QML) [1] is to take advantage of quantum computers to address learning tasks such as classification, regression, and content generation. A key component in QML is the data encoding strategy: the method by which classical data is embedded into the initial quantum states, to be subsequently manipulated using hybrid learning algorithms [2, 3]. The encoding step is, in essence, the most important one, as it defines the working principle of the quantum algorithm, often constitutes the bottleneck step in terms of runtime, and the choice of encoding significantly influences the performance of the learning algorithm.

Commonly used encoding schemes, such as angle encoding and amplitude encoding [2], are not efficient. Angle encoding maps classical data $x = (x_1, ..., x_n)$ into a quantum state using rotation angles:

$$|x\rangle = \bigotimes_{i=1}^{n} \cos(x_i)|0\rangle + \sin(x_i)|1\rangle. \tag{1}$$

However, angle encoding is not efficient in terms of the number of qubits, as it can encode only one feature per qubit. Amplitude encoding, instead, maps a normalized classical N-dimensional data x into the amplitudes of a quantum state:

$$|\psi_x\rangle = \sum_{i=1}^{N} x_i|i\rangle \tag{2}$$

where $N = 2^n$, x_i is the ith element of x and $|i\rangle$ is the ith computational basis state. However, amplitude encoding requires a quantum circuit whose depth increases exponentially with the number of qubits [4, 5, 6]. This is a relevant issue, as in the current Noisy Intermediate-Scale Quantum (NISQ) era, quantum devices are prone to noise and deeper circuits tend to produce noise. Furthermore, these methods do not take advantage of the structure of the data, which could be exploited for improved classification or generalization.

✉ marco.mordacci1@unipr.it (M. Mordacci); mahul.pandey@unipr.it (M. Pandey); paolo.santini@unipr.it (P. Santini); michele.amoretti@unipr.it (M. Amoretti)

ⓘD 0009-0001-1955-7197 (M. Mordacci); 0009-0009-6654-2746 (M. Pandey); 0000-0002-1182-0173 (P. Santini); 0000-0002-6046-1904 (M. Amoretti)

In this work, a novel approach to quantum encoding is explored, specifically aimed at classification problems, that is adaptive and data-driven. Instead of using an a-priori fixed encoding map, an encoding circuit is designed and optimized specifically for the classification task at hand. More precisely, it aims to embed input data in a way that explicitly achieves class separability in the Hilbert space. The method is inspired by FaceNet [7], a classical deep learning model for face recognition that employs a triplet loss function to encourage similar inputs to be mapped close together and dissimilar ones far apart in the embedding space.

The paper is organized as follows. In Section 2, the state-of-the-art is discussed and some of the recent developments in addressing the problem of quantum embeddings for machine learning are summarized. In Section 3, the mathematical problem and the objective are defined. The strategy is described in Section 4, where the triplet loss and how it drives the embedding circuit construction are explained. The basic variational classifier that uses these embedded states to make predictions is also described in that section. In Section 5, results from both simulation and real hardware are reported. Finally, in Section 6 an overview of findings and future outlook is presented.

2. Related Work

In recent years, the problem of encoding classical data into a quantum state has attracted considerable interest. A well-designed encoding scheme can simplify the task for the Variational Quantum Circuit (VQC), but the most commonly used methods, angle and amplitude encoding, present several challenges. Several algorithms have been developed in recent years to address this problem.

A comprehensive benchmarking of the most well-known encoding schemes, viz. amplitude, angle and IQP embedding [8] has been carried out in [9] on various datasets including the MNIST01 and MNIST08 subsets. These results are useful as a benchmark test for advanced encoding schemes that better take into account the data structure.

In recent years, a research field known as Quantum Architecture Search (QAS) [10] has emerged, aiming to identify the optimal quantum circuit architecture for both the encoding stage and the VQC. Techniques such as Reinforcement Learning [11, 12], evolutionary algorithms [13, 14, 15, 16], and particle swarm optimization were applied [12, 17]. Genetic algorithms [18] are used to learn the whole architecture of the encoding scheme. Those are optimization techniques based on the theory of evolution. The algorithms evolve a population of individuals with the encoded feature maps, through the application of genetic operators. In each generation, the resulting offspring is selected in order to improve the objectives. In [15], the authors optimized the quantum feature maps in a quantum kernel Support Vector Machine (SVM) [19]. The genetic algorithm stores the circuit of each individual and aims to maximize the accuracy of the feature maps in modeling the data, while minimizing circuit complexity. The results show that it can produce a classifier with 100% accuracy that generalizes well to unseen data.

In [20], the authors, inspired by the so-called classical "metric learning" [21, 22], trained the quantum embedding using the trace and Hilbert-Schmidt distances.

Drawing inspiration from classical kernel methods [23], quantum kernel methods map classical data into a high-dimensional quantum feature space [24, 25, 26, 27]. The similarity between data points is then calculated using a quantum kernel, which is essentially the inner product of the quantum states. In [28], two strategies inspired by kernel theory [23] to find patterns in the data were proposed. One estimates intractable quantum kernels by feeding them into a classical kernel method; the other applies VQC to learn models that process the feature vectors.

Rath et al. [29] analyzed the impact of quantum encoding techniques (basis, superposition, angle, and amplitude encodings) in classical machine learning. Various classical algorithms were tested, such as SVM, Decision Tree, Random Forest, and AdaBoost.

In [30], an efficient method for amplitude encoding of real polynomial functions was proposed.

Another interesting embedding is the Hamiltonian encoding [2, 31], where classical data are encoded into the parameters of a system's Hamiltonian.

The choice of the encoding method plays a crucial role, as it can lead to the emergence of barren plateaus [32], which cause the gradients to vanish exponentially with the size of the system, hindering effective training. The entanglement entropy [33] generated by a quantum circuit can follow either a volume law or an area law. In a volume law regime, entanglement scales proportionally with the total number of qubits in the system; while in the area law scenario, entanglement entropy scales proportionally with the number of qubits at the boundary. Consequently, the applied encoding method can induce barren plateaus by generating excessive entanglement entropy [34].

3. Problem Statement

The goal of the data encoding strategy is to design an encoding that is well-suited for complex classification tasks involving unstructured data, such as image classification. As a concrete test case, a binary image classification task is considered using both MNIST and MedMNIST datasets [35]. As a simple starting point, the datasets are restricted to the first two classes (e.g., "0" vs "1"). An embedding circuit $U_{emb}(\mathbf{x}^{(j)})$ is constructed (where $\mathbf{x}^{(j)}$ are the sets of feature vectors in the training dataset), such that the resulting state vectors $|\psi^{(j)}\rangle = U_{emb}(\mathbf{x}^{(j)})|0\rangle$ lie in compact and mutually orthogonal subspaces of the Hilbert space depending on their class labels $y^{(j)}$. For an ideal encoding, the following should be obtained:

$$\langle \psi^{(i)}|\psi^{(j)}\rangle \simeq \begin{cases} 1, & y^{(i)} = y^{(j)} \\ 0, & y^{(i)} \neq y^{(j)}. \end{cases} \tag{3}$$

Of course, an ideal encoding is not possible for such a complex, unstructured dataset; therefore, the algorithm aims to cluster the encoded state vectors into distinct regions of the Hilbert space. To introduce a natural metric on the space of states and to allow for future generalizations to noisy and partial measurements, density matrices $\rho^{(i)}$ are taken into account rather than state vectors. A natural metric on the Hilbert space is then the trace distance:

$$D(\rho_1, \rho_2) = \frac{1}{2} \text{tr}\sqrt{(\rho_1 - \rho_2)^\dagger (\rho_1 - \rho_2)}. \tag{4}$$

Two sets of pairs are defined:

- Intra-class pairs $\mathscr{P}_{same} = \{(i,j)|y^{(i)} = y^{(j)}\}$,
- Inter-class pairs $\mathscr{P}_{diff} = \{(i,j)|y^{(i)} \neq y^{(j)}\}$.

The objective is to minimize the average distance $\mathbb{E}_{(i,j)\in\mathscr{P}_{same}}D(\rho^{(i)}, \rho^{(j)})$ while maximizing the average distance $\mathbb{E}_{(i,j)\in\mathscr{P}_{diff}}D(\rho^{(i)}, \rho^{(j)})$.

4. Proposed Solution

A triplet-loss-driven construction of a quantum encoding circuit that maps classical images to expressive quantum states is proposed. The entire algorithm proceeds in three main stages:

1. Triplet selection,
2. Greedy circuit construction,
3. Variational classification circuit.

4.1. Triplet selection

A strategy similar to the hard mining strategy in classical image recognition [7] is adopted. For each ordered pair of classes (a, b), the following is chosen:

- An anchor A: The median of all image vectors belonging to class a.

- A positive P: The image vector in class a that is the farthest from the anchor (hard positive).
- A negative N: The image vector in class b that is the closest to the anchor (hard negative).

The goal is to consider the worst-case scenario, selecting the farthest image of the same class as the anchor (positive) and the closest image from the other class (negative), in order to minimize the distance between the anchor and the positive, while maximizing the distance between the anchor and the negative.

This triplet is designed to be maximally informative, promoting embeddings that tightly cluster class-relevant features while pushing away confusing examples. To demonstrate, the simple problem of binary classification with only digits "0" and "1" of the MNIST dataset is considered. There are only two triplets corresponding to $(0, 1)$ and $(1, 0)$ respectively. The triplets are shown in Figure 1.
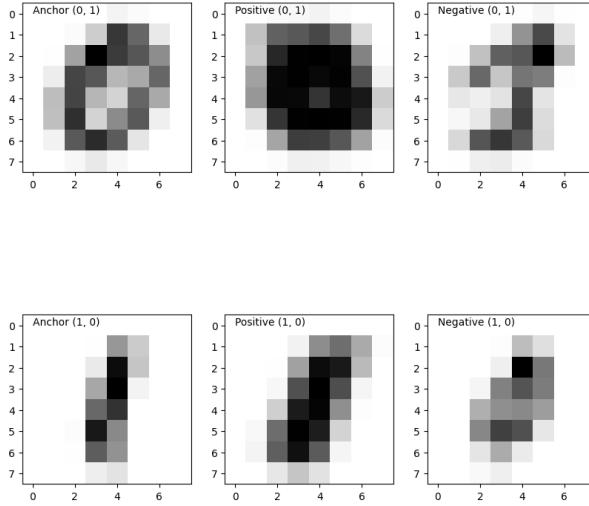
Figure 1: Triplets for the binary (0,1) subset of the MNIST dataset. The image resolution is 8×8.

4.2. Greedy Circuit Construction

Two hyperparameters are introduced:

- A weight factor w that introduces a relative weight between the positive and negative loss.
- A margin m to adjust the degree of separation between the different clusters.

The following objective function is minimized:

$$J = \sum_{triplets} \max(0, wD(A, P) - D(A, N) + m). \tag{5}$$

This is performed with a greedy combinatorial optimization algorithm:

1. Start with an empty circuit.
2. At each step i, pick the ith feature (pixel value) to encode.
3. Incrementally add gates from a fixed pool: $\{R_X, R_Y, R_Z, CNOT, CZ\}$ at each step as follows:
   a) Go through all possible gate-qubit combinations. For single-qubit rotations, the feature vector x_i is encoded in the rotation angle.
   b) Temporarily append each combination to the previously constructed circuit, and evaluate the density matrices corresponding to each triplet element.
   c) Compute the triplet loss for each choice.
   d) Pick the gate-qubit combination that corresponds to the lowest triplet loss (greedy choice).

e) If this best choice of gate turns out to be a *CNOT* or a *CZ*, make sure to follow it up by a rotation gate that best encodes the ith feature. Thus, this feature is actually being encoded by an entangling gate followed by a rotation.
4. Repeat this until all features have been encoded or a target depth is reached.
5. Store the sequence of gate-qubit combinations: this serves as embedding circuit U_{emb}.

The controlled rotations (CR_x, CR_y, CR_z) are not included in the set of possible gates, as the optimal solutions that minimize the loss never select them. Instead, *CNOT* gates and single-qubit rotations consistently perform better.

4.3. Variational Classification Layer

A trainable parameterized quantum circuit $U_{var}(\theta)$ is applied to the output of $U_{emb}(x)$, forming a hybrid quantum model. The parameters θ are optimized using a standard classical optimizer (Adam) to minimize the classification loss (cross-entropy).

The VQC of [9] is used, each layer of which consists of:

- A layer of R_Y rotations, each parametrized by $\{\theta_1, ...\theta_n\}$ applied to each qubit.
- A circular layer of CNOTs.

For a binary classification problem, the last qubit is measured and the probability of it being in the state $|1\rangle$ is obtained. One layer of the VQC is visualized in Fig. 2.
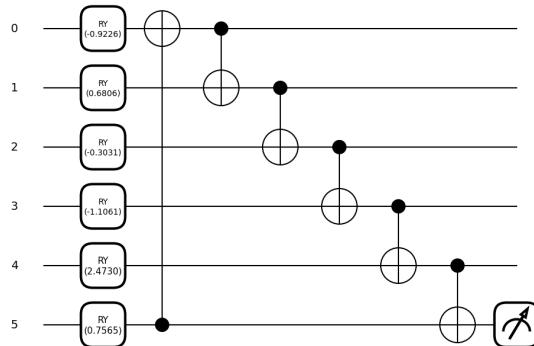
Figure 2: One layer of the variational circuit for a random choice of parameters.

Performance is evaluated using only one or two layers of the VQC, to demonstrate the improvement over the usual amplitude encoding scheme.

5. Experimental Evaluation

5.1. Preprocessing

Images are rescaled to $d \times d$ pixels, such that $n_{features} = d^2$. The proposed encoding scheme works better if no further rescaling is applied to the data set.

5.2. Hyperparameter Selection

For a smaller scale of the problem, it is sufficient to set the margin parameter in (5) to $m = 0$. Also, for best results, it helps to choose a starting weight of $w = 1.0$ (which gives equal weightage to both cluster separation and condensation within the cluster), and dampening it by a small amount (0.01) at each

step. The choice of using a cut-off of 0 for the loss function does not significantly affect the results, so the following objective is used:

$$J = - \sum_{triplets} (D(A, N) - wD(A, P)) \qquad (6)$$

with w receiving a small damping at each iteration.

N_{qubits} is set to $\log_2 n_{features}$ since a lower number would lead to deeper circuits, worsen performance and noise resilience, and could lead to barren plateau problems.

For the benefit of the reader, a typical embedding circuit generated for 8-by-8 MNIST01 images is presented in Figure 3, with the above selection of hyperparameters. The features are encoded in the rotation angles of the single-qubit rotation gates. Note that the total gate count in this circuit is less than 64, because consecutive rotations have been merged into a single rotation. At the end of the circuit, all qubits are measured in order to retrieve the density matrices and compute the trace distances. When the encoding circuit is ready, it will be followed by the VQC, without any measurement.
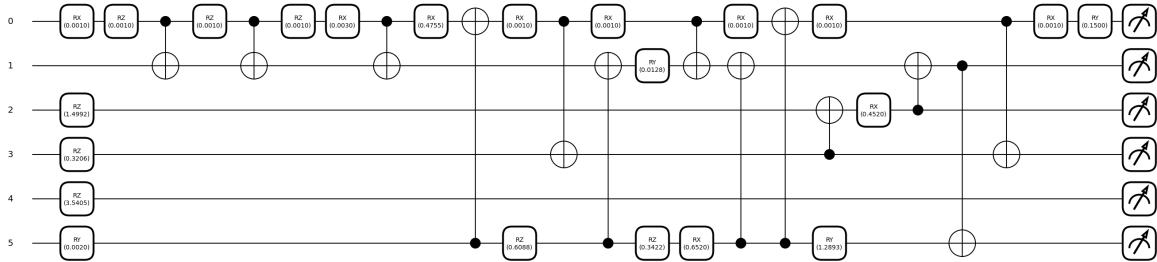
Figure 3: A typical embedding circuit. We have used the encoding of a 0 image as an example.

5.3. MNIST Results

The clusters of the encoded density matrix are shown in Figure 4, with the help of a t-SNE embedding, for image resolutions ranging from 8-by-8 to 12-by-12 pixels. This demonstrates that the embedding performs quite well in cluster separation even with considerably low image resolutions.

In Figure 5, the epoch-wise accuracy is plotted while training the VQC layers for 8-by-8 images (64 features), with the proposed encoding scheme and amplitude encoding, respectively. The proposed encoding matches or surpasses amplitude encoding in the accuracy of results for low resolution of images, while being more efficient in terms of circuit depth.

The various metrics across feature size are shown in Table 1. When a single layer of the VQC is used, the proposed method outperforms amplitude encoding (except in the 8-qubit case). However, with a more expressive two-layer VQC, the difference between the two encoding methods diminishes, and the choice of encoding circuit becomes less critical. Nonetheless, using fewer gates remains advantageous for execution on real devices, as it reduces noise, an effect that will be demonstrated in Section 5.5. In fact, the number of gates needed in the proposed method is much lower than those applied in amplitude encoding. Indeed, if the Mottonen state preparation [4] is applied to implement the amplitude embedding, the CNOT and rotation counts are:

- 6 qubits: 228 CNOT, 251 Rotations,
- 7 qubits: 480 CNOT, 507 Rotations,
- 8 qubits: 980 CNOT, 1019 Rotations.

On the contrary, the total gate count of the proposed method is:

- 64 features (6 qubits): 13 CNOT, 64 rotations at most (one for each feature, but consecutive rotations can be merged),
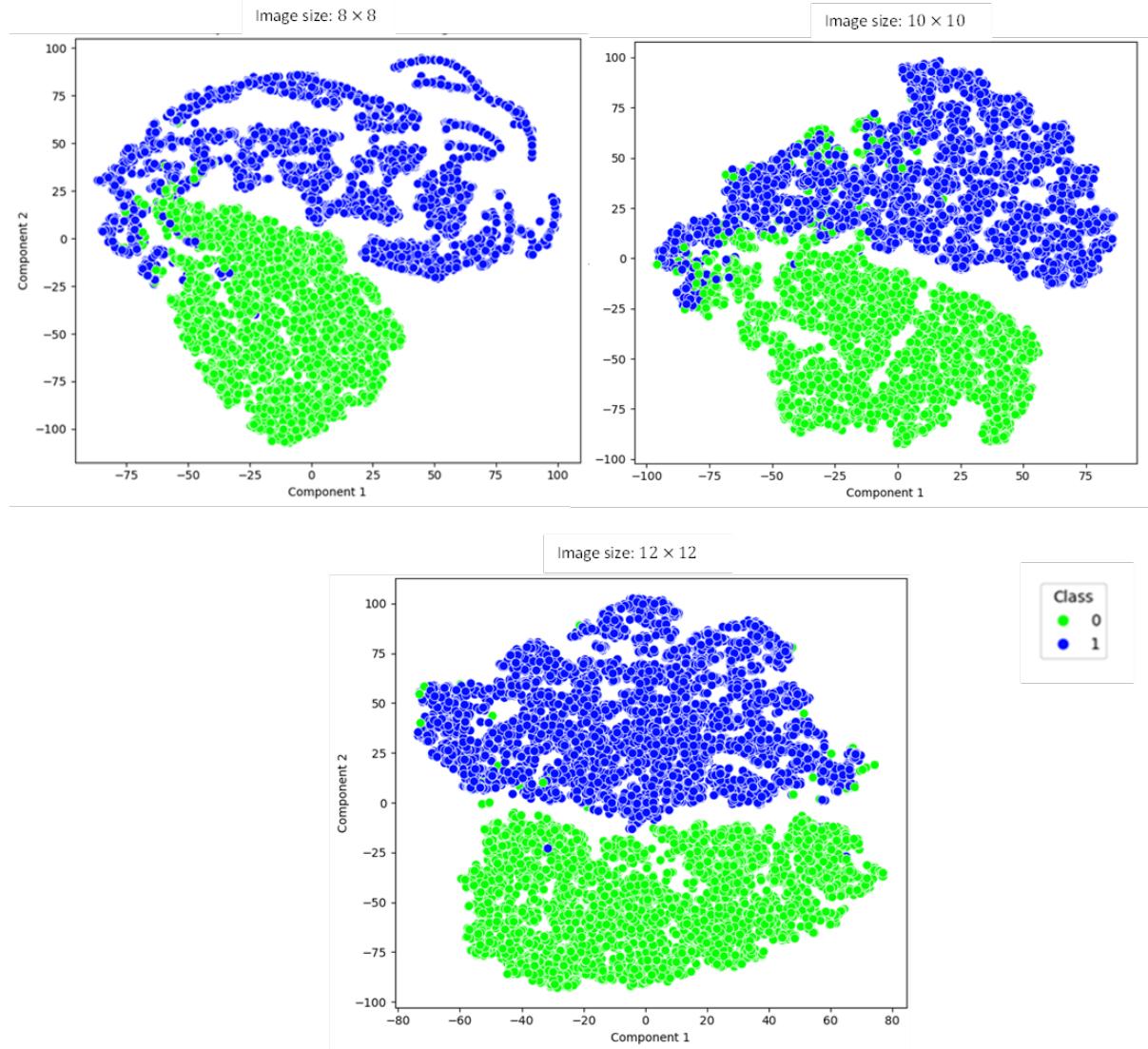
Figure 4: A visualization of the embedded density matrices using t-SNE, across different system sizes.
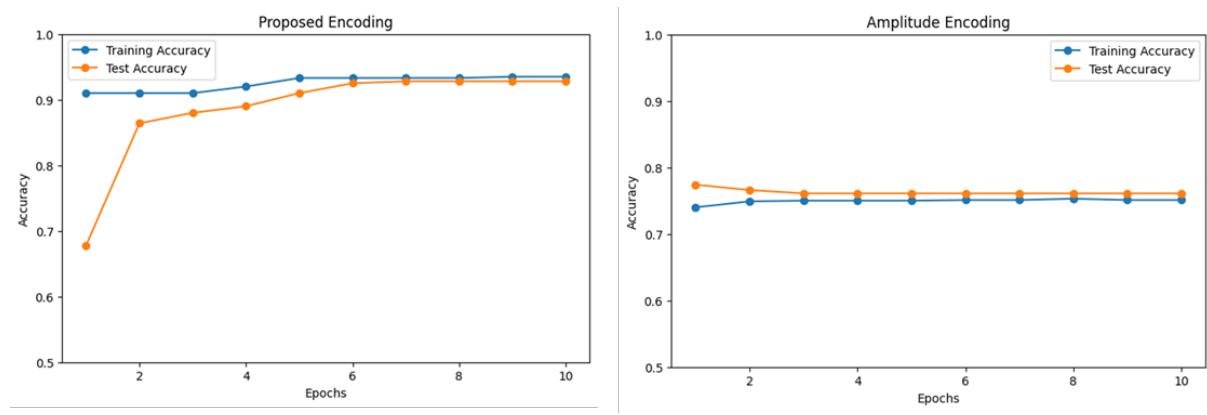

Figure 5: Epoch-wise accuracy curves during training of VQC with 1 layer, for 8×8 MNIST images.

- 100 features (7 qubits): 14 CNOT, 100 rotations at most,
- 144 features (8 qubits): 24 CNOT, 144 rotations at most,
- 256 features (8 qubits): 47 CNOT, 256 rotations at most.

Note that amplitude encoding requires padding of the feature vector to match a power of 2, but the proposed method does not require such padding.

$N_{features}$	N_{qubits}	N_{layers}	Encoding	Accuracy	class	Precision	Recall	f1-score
8×8	6	1	Amplitude	77%	0 1	0.67 0.97	0.98 0.59	0.80 0.73
			Proposed	93%	0 1	0.96 0.91	0.89 0.97	0.92 0.94
		2	Amplitude	97%	0 1	0.97 0.97	0.98 0.96	0.97 0.97
			Proposed	97%	0 1	0.97 0.97	0.96 0.98	0.97 0.97
10×10	7	1	Amplitude	62%	0 1	0.58 0.68	0.70 0.56	0.63 0.61
			Proposed	91%	0 1	0.90 0.92	0.91 0.91	0.91 0.92
		2	Amplitude	87%	0 1	0.78 0.99	0.99 0.76	0.87 0.86
			Proposed	92%	0 1	0.93 0.90	0.90 0.94	0.91 0.92
12×12	8	1	Amplitude	94%	0 1	0.90 0.98	0.98 0.91	0.94 0.94
			Proposed	97%	0 1	0.96 0.97	0.96 0.97	0.96 0.97
		2	Amplitude	97.5%	0 1	0.97 0.98	0.97 0.97	0.97 0.97
			Proposed	97%	0 1	0.98 0.97	0.96 0.98	0.97 0.97
16×16	8	1	Amplitude	95%	0 1	0.90 1.0	1.0 0.91	0.95 0.94
			Proposed	54%	0 1	0.50 0.62	0.74 0.36	0.60 0.46
		2	Amplitude	95%	0 1	0.95 0.94	0.94 0.95	0.95 0.94
			Proposed	84%	0 1	0.86 0.83	0.79 0.89	0.82 0.86

Table 1
Comparison of metrics for different feature size

5.4. MedMNIST Results

In this case, a learning rate of 0.01 and a batch size of 32 are used. For each result, the training is executed 5 times and the mean results are taken. For the Tissue and OCT datasets, a smaller training set of 3500 samples (1750 samples for each class) is used to reduce the encoding circuit generation time. The results obtained with the proposed method are compared to those achieved using amplitude encoding. Only one layer of the circuit shown in Fig. 2 is used.

Table 2 reports the test set accuracies on various MedMNIST datasets, where the images are rescaled to 8×8. The class-wise precision, recall, and f1-score of a random execution are presented. An improvement in performance cannot be observed. In fact, both methods tend to predict a single class for all examples, indicating a failure to learn meaningful patterns. The only improvement is visible in the organA dataset; while amplitude encoding still predicts a single class, the proposed method begins to learn.

However, with 16×16 images, the differences between the two encoding methods become visible, as shown in Table 3. Indeed, for Chest, Breast, and OrganC datasets, the proposed encoding method

$N_{features}$	N_{qubits}	Dataset	Encoding	Accuracy	class	Precision	Recall	f1-score
8 × 8	6	Chest	Amplitude	54%	0	0.54	0.65	0.59
					1	0.49	0.38	0.43
			Proposed	53%	0	0.53	1.0	0.69
					1	0.55	0.001	0.002
		Breast	Amplitude	73%	0	0.0	0.0	0.0
					1	0.73	1.0	0.84
			Proposed	73%	0	0.0	0.0	0.0
					1	0.73	1.0	0.84
		oct	Amplitude	55%	0	0.59	0.32	0.41
					1	0.53	0.78	0.64
			Proposed	50%	0	0.0	0.0	0.0
					1	0.50	1.0	0.67
		Tissue	Amplitude	44%	0	0.89	0.39	0.54
					1	0.14	0.67	0.23
			Proposed	87%	0	0.87	1.0	0.93
					1	0.0	0.0	0.0
		Pneumonia	Amplitude	62%	0	0.0	0.0	0.0
					1	0.62	1.0	0.77
			Proposed	62%	0	0.0	0.0	0.0
					1	0.62	1.0	0.77
		OrganA	Amplitude	58.5%	0	0.58	1.0	0.72
					1	0.0	0.0	0.0
			Proposed	**67%**	0	**0.78**	**0.63**	**0.70**
					1	**0.61**	**0.76**	**0.68**
		OrganC	Amplitude	66%	0	0.66	1.0	0.79
					1	0.0	0.0	0.0
			Proposed	66%	0	0.66	1.0	0.79
					1	0.0	0.0	0.0
		OrganS	Amplitude	65%	0	0.65	1.0	0.79
					1	0.0	0.0	0.0
			Proposed	65%	0	0.65	1.0	0.79
					1	0.0	0.0	0.0

Table 2

8 × 8 MedMNIST results on the test set with different dataset.

performs similarly to the amplitude encoding in terms of accuracy; on the other hand, amplitude encoding performs better with the Tissue and OCT datasets, due to the fact that the smaller training set loses the true member of the triplet for each class. However, for the Tissue dataset, the proposed encoding method starts recognizing some members of the positive class, even though in the test set there are more negative than positive examples. Furthermore, the proposed method achieves better accuracy with the Pneumonia, OrganA, and OrganS datasets. In this case, while amplitude encoding does not learn meaningful patterns and still predicts the same class every time (such as in the 8 × 8 scenario), the proposed encoder is starting to learn both classes.

With 28 × 28 images; the proposed encoding shows a larger improvement in recognizing both classes, as shown in Table 4. Now, it achieves better results also on the OrganC and the Tissue datasets and, in general, gives a more balanced f1-score across classes than the amplitude encoding.

In Figure 6, an example of training on the OrganA validation set is presented.

Furthermore, the MNIST dataset is trained with only 3500 samples to understand how the smaller part of the training set affects the results for the Tissue and OCT datasets. The proposed method achieves an accuracy of 75%, while the amplitude encoding reaches approximately 80%. These results show a performance deterioration due to the absence of the original anchor, positive, and negative images in the reduced training set. Therefore, other techniques will be explored to reduce the computational time.

$N_{features}$	N_{qubits}	Dataset	Encoding	Accuracy	Class	Precision	Recall	f1-score
16 × 16	8	Chest	Amplitude	52.5%	0	0.55	0.63	0.59
					1	0.50	0.41	0.45
			Proposed	52%	0	0.55	0.64	0.59
					1	0.49	0.40	0.44
		Breast	Amplitude	62%	0	0.38	0.71	0.50
					1	0.85	0.58	0.69
			Proposed	53%	0	0.30	0.57	0.40
					1	0.77	0.52	0.62
		oct	Amplitude	54%	0	0.54	0.57	0.56
					1	0.54	0.51	0.53
			Proposed	50%	0	0.75	0.01	0.02
					1	0.50	1.00	0.67
		Tissue	Amplitude	66%	0	0.87	0.71	0.78
					1	0.12	0.27	0.17
			Proposed	82%	0	0.87	0.93	0.90
					1	0.11	0.05	0.07
		Pneumonia	Amplitude	62%	0	0.0	0.0	0.0
					1	0.62	1.0	0.77
			Proposed	**59%**	0	**0.46**	**0.76**	**0.57**
					1	**0.76**	**0.47**	**0.58**
		OrganA	Amplitude	58.5%	0	0.58	1.0	0.72
					1	0.0	0.0	0.0
			Proposed	**60%**	0	**0.6**	**0.90**	**0.72**
					1	**0.6**	**0.19**	**0.29**
		OrganC	Amplitude	66%	0	0.66	1.0	0.79
					1	0.0	0.0	0.0
			Proposed	66%	0	0.66	1.0	0.79
					1	0.0	0.0	0.0
		OrganS	Amplitude	65%	0	0.65	1.0	0.79
					1	0.0	0.0	0.0
			Proposed	**58%**	0	**0.64**	**0.88**	**0.74**
					1	**0.32**	**0.10**	**0.16**

Table 3

16 × 16 MedMNIST results on the test set with different dataset.
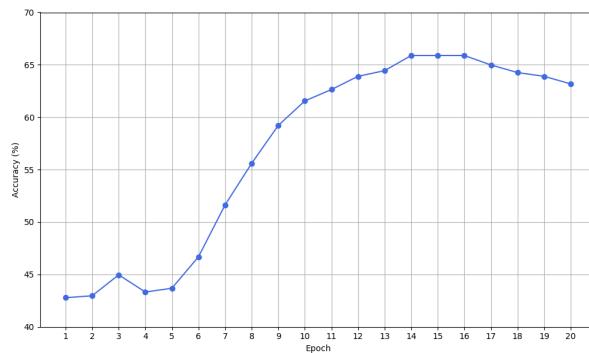

Figure 6: Accuracy on 8 × 8 organA validation dataset during training.

5.5. Real Hardware Execution

The produced encoding circuit with the trained VQC for 8×8 MNIST images is tested on Rigetti Ankaa-3 and IQM Garnet quantum computers, through AWS Braket. Each circuit is executed with 1024 shots. The first 500 samples of the test set are taken into account. In Table 5 the accuracy, precision, recall, and

$N_{features}$	N_{qubits}	Dataset	Encoding	Accuracy	Class	Precision	Recall	f1-score
28×28	10	Chest	Amplitude	54%	0	0.54	0.96	0.69
					1	0.60	0.06	0.11
			Proposed	52%	0	0.54	0.68	0.60
					1	0.48	0.33	0.40
		Breast	Amplitude	49%	0	0.24	0.43	0.31
					1	0.71	0.51	0.59
			Proposed	46%	0	0.19	0.31	0.24
					1	0.67	0.52	0.59
		oct	Amplitude	51%	0	0.52	0.37	0.43
					1	0.51	0.65	0.57
			Proposed	50%	0	0.50	0.48	0.49
					1	0.50	0.52	0.51
		Tissue	Amplitude	38%	0	0.88	0.34	0.49
					1	0.13	0.69	0.22
			Proposed	**68%**	0	**0.87**	**0.75**	**0.80**
					1	**0.12**	**0.23**	**0.15**
		Pneumonia	Amplitude	62%	0	0.0	0.0	0.0
					1	0.62	1.0	0.77
			Proposed	**55%**	0	**0.42**	**0.44**	**0.43**
					1	**0.66**	**0.64**	**0.65**
		OrganA	Amplitude	58.5%	0	0.58	1.0	0.72
					1	0.0	0.0	0.0
			Proposed	**63%**	0	**0.66**	**0.74**	**0.69**
					1	**0.58**	**0.49**	**0.53**
		OrganC	Amplitude	66%	0	0.66	1.0	0.79
					1	0.0	0.0	0.0
			Proposed	**54%**	0	**0.69**	**0.53**	**0.60**
					1	**0.37**	**0.54**	**0.44**
		OrganS	Amplitude	65%	0	0.65	1.0	0.79
					1	0.0	0.0	0.0
			Proposed	**58%**	0	**0.70**	**0.65**	**0.67**
					1	**0.43**	**0.49**	**0.46**

Table 4

28×28 MedMNIST results on the test set with different dataset.

f1-score are reported. On Rigetti Ankaa-3, the results are worse due to noise, as the encoding circuit still has a high depth, even though the number of rotations and CNOT gates is lower than in the amplitude encoding. However, on IQM Garnet, the noise does not affect the results. The accuracy is identical to that obtained in simulation; there are only minor variations in precision, recall, and f1-score.

$n_{features}$	N_{qubits}	Hardware	Accuracy	Class	Precision	Recall	f1-score
8×8	6	Rigetti Ankaa-3	50%	0	0.33	0.14	0.19
				1	0.78	0.54	0.64
		IQM Garnet	96.8%	0	0.97	0.95	0.96
				1	0.97	0.98	0.97
		Simulation	96.8%	0	0.99	0.93	0.96
				1	0.95	0.99	0.97

Table 5

Comparison of accuracies on simulation and real hardware during the testing of the MNIST dataset.

6. Conclusion

In this work, a novel quantum encoding scheme based on triplet loss is proposed. Specifically, the gates applied to the encoding circuit are selected to maximize the distance between different classes and minimize the distance between samples of the same class. The generated encoding circuit has a lower depth than amplitude encoding while using the same number of features, and it does not lose any information, unlike angle encoding, which uses only a few features to encode the data.

This encoding scheme is tested on various binary classification tasks. The MNIST dataset and various MedMNIST datasets are taken into account to evaluate the encoding performance. Furthermore, the method is also evaluated on real quantum hardware to determine the impact of noise on the results.

The proposed encoding scheme is very successful in class separation for simple and distinctive image datasets like the MNIST dataset, and for low-resolution images achieves better results than amplitude encoding with a much lesser circuit depth. For more complex image datasets like the MedMNIST, where variations between images of different classes are much harder to spot, the encoding scheme performs more poorly when low-resolution images are used; however, at high enough resolutions, it starts to capture these inter-class differences better.

Future work will involve modifying the encoding generator to further enhance performance. In particular, the current "hard" selection of triplets is too simplistic, and more complex image datasets are likely to benefit from a more effective triplet-mining strategy. Additionally, the "greedy" algorithm for adding gates may be refined to incorporate strategies that better capture correlations among different features and to reduce the encoding circuit generation time, as computational time increases with higher-resolution images or larger datasets.

Currently, when different gate configurations applied to different qubits yield the same optimal loss, the gates acting on the first qubits are preferentially selected. As a result, a disproportionately high number of operations is applied to the first qubits, leading to an increased circuit depth, and making the other qubits redundant. This issue will be addressed in the future.

Furthermore, an evaluation of deeper VQC will be performed, and the algorithm will also be tested in multiclass scenarios.

Acknowledgment

Declaration on Generative AI

During the preparation of this work, the authors used ChatGPT in order to: Grammar and spelling check, Paraphrase and reword. After using this tool, the authors reviewed and edited the content as needed and takes full responsibility for the publication's content.

References

[1] J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe, S. Lloyd, Quantum machine learning, Nature 549 (2017) 195–202.
[2] M. Schuld, F. Petruccione, Supervised learning with quantum computers, volume 17, Springer, 2018.
[3] R. LaRose, B. Coyle, Robust data encodings for quantum classifiers, Physical Review A 102 (2020) 032420.

[4] M. Mottonen, J. J. Vartiainen, V. Bergholm, M. M. Salomaa, Transformation of quantum states using uniformly controlled rotations, arXiv preprint quant-ph/0407010 (2004).

[5] X. Sun, G. Tian, S. Yang, P. Yuan, S. Zhang, Asymptotically optimal circuit depth for quantum state preparation and general unitary synthesis, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 42 (2023) 3301–3314.

[6] G. Belli, M. Mordacci, M. Amoretti, Srbb-based quantum state preparation, in: Proceedings of the 22nd ACM International Conference on Computing Frontiers, 2025, pp. 172–175.

[7] F. Schroff, D. Kalenichenko, J. Philbin, Facenet: A unified embedding for face recognition and clustering, in: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 815–823.

[8] V. Havlíček, A. D. Córcoles, K. Temme, A. W. Harrow, A. Kandala, J. M. Chow, J. M. Gambetta, Supervised learning with quantum-enhanced feature spaces, Nature 567 (2019) 209–212. URL: http://dx.doi.org/10.1038/s41586-019-0980-2. doi:10.1038/s41586-019-0980-2.

[9] O. Zang, G. Barrué, T. Quertier, Benchmarking data encoding methods in quantum machine learning, 2025. URL: https://arxiv.org/abs/2505.14295. arXiv:2505.14295.

[10] D. Martyniuk, J. Jung, A. Paschke, Quantum architecture search: a survey, in: 2024 IEEE International Conference on Quantum Computing and Engineering (QCE), volume 1, IEEE, 2024, pp. 1695–1706.

[11] E.-J. Kuo, Y.-L. L. Fang, S. Y.-C. Chen, Quantum architecture search via deep reinforcement learning, arXiv preprint arXiv:2104.07715 (2021).

[12] M. Kölle, D. Seidl, M. Zorn, P. Altmann, J. Stein, T. Gabor, Optimizing variational quantum circuits using metaheuristic strategies in reinforcement learning, in: 2024 IEEE International Conference on Quantum Computing and Engineering (QCE), volume 2, IEEE, 2024, pp. 323–328.

[13] A. Jin, T. Dutta, M. Mukherjee, J. Latorre, et al., Practicality of training a quantum machine in the nisq era, in: APS Division of Atomic, Molecular and Optical Physics Meeting Abstracts, volume 2024, 2024, pp. D00–081.

[14] H. Wang, Y. Ding, J. Gu, Y. Lin, D. Z. Pan, F. T. Chong, S. Han, Quantumnas: Noise-adaptive search for robust quantum circuits, in: 2022 IEEE International Symposium on High-Performance Computer Architecture (HPCA), IEEE, 2022, pp. 692–708.

[15] S. Altares-López, A. Ribeiro, J. J. García-Ripoll, Automatic design of quantum feature maps, Quantum Science and Technology 6 (2021) 045015.

[16] L. Sünkel, D. Martyniuk, D. Mattern, J. Jung, A. Paschke, Ga4qco: genetic algorithm for quantum circuit optimization, arXiv preprint arXiv:2302.01303 (2023).

[17] M. Mordacci, M. Amoretti, Training variational quantum circuits using particle swarm optimization, in: 2025 IEEE International Conference on Quantum Computing and Engineering (QCE), IEEE, 2025.

[18] S. Katoch, S. S. Chauhan, V. Kumar, A review on genetic algorithm: past, present, and future, Multimedia tools and applications 80 (2021) 8091–8126.

[19] P. Rebentrost, M. Mohseni, S. Lloyd, Quantum support vector machine for big data classification, Physical review letters 113 (2014) 130503.

[20] S. Lloyd, M. Schuld, A. Ijaz, J. Izaac, N. Killoran, Quantum embeddings for machine learning, arXiv preprint arXiv:2001.03622 (2020).

[21] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, R. Shah, Signature verification using a" siamese" time delay neural network, Advances in neural information processing systems 6 (1993).

[22] S. Chopra, R. Hadsell, Y. LeCun, Learning a similarity metric discriminatively, with application to face verification, in: 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05), volume 1, IEEE, 2005, pp. 539–546.

[23] T. Hofmann, B. Schölkopf, A. J. Smola, Kernel methods in machine learning (2008).

[24] M. Mercadier, Quantum-enhanced versus classical support vector machine: An application to stock index forecasting, Available at SSRN 4630419 (2023).

[25] G. Gentinetta, A. Thomsen, D. Sutter, S. Woerner, The complexity of quantum support vector machines, Quantum 8 (2024) 1225.

[26] Z. Yin, I. Agresti, G. de Felice, D. Brown, A. Toumi, C. Pentangelo, S. Piacentini, A. Crespi, F. Ceccarelli, R. Osellame, et al., Experimental quantum-enhanced kernels on a photonic processor, arXiv preprint arXiv:2407.20364 (2024).

[27] C. Ding, S. Wang, Y. Wang, W. Gao, Quantum machine learning for multiclass classification beyond kernel methods, Physical Review A 111 (2025) 062410.

[28] M. Schuld, N. Killoran, Quantum machine learning in feature hilbert spaces, Physical review letters 122 (2019) 040504.

[29] M. Rath, H. Date, Quantum data encoding: A comparative analysis of classical-to-quantum mapping techniques and their impact on machine learning accuracy, EPJ Quantum Technology 11 (2024) 72.

[30] J. Gonzalez-Conde, T. W. Watts, P. Rodriguez-Grasa, M. Sanz, Efficient quantum amplitude encoding of polynomial functions, Quantum 8 (2024) 1297.

[31] O. Di Matteo, A. McCoy, P. Gysbers, T. Miyagi, R. Woloshyn, P. Navrátil, Improving hamiltonian encodings with the gray code, Physical Review A 103 (2021) 042405.

[32] M. Larocca, S. Thanasilp, S. Wang, K. Sharma, J. Biamonte, P. J. Coles, L. Cincio, J. R. McClean, Z. Holmes, M. Cerezo, Barren plateaus in variational quantum computing, Nature Reviews Physics (2025) 1–16.

[33] P. Calabrese, J. Cardy, Entanglement entropy and quantum field theory, Journal of statistical mechanics: theory and experiment 2004 (2004) P06002.

[34] L. Leone, S. F. Oliviero, L. Cincio, M. Cerezo, On the practical usefulness of the hardware efficient ansatz, Quantum 8 (2024) 1395.

[35] J. Yang, R. Shi, B. Ni, Medmnist classification decathlon: A lightweight automl benchmark for medical image analysis, in: 2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI), IEEE, 2021, pp. 191–195.