Int. J. Appl. Math. Comput. Sci., 2025, Vol. 35, No. 4, 651–666
DOI: 10.61822/amcs-2025-0046

A QUANTUM CONVOLUTION AUTOENCODER FOR HANDWRITTEN LETTERS AND DIGITS: A CASE STUDY

MAREK SAWERWAIN [a,*], MAREK KOWAL [a], JÓZEF KORBICZ [a]

[a]Institute of Control and Computation Engineering
University of Zielona Góra
ul. Szafrana 2, 65-516 Zielona Góra, Poland
e-mail: {M.Sawerwain,M.Kowal,J.Korbicz}@issi.uz.zgora.pl

The concept of an autoencoder is a key element of modern machine learning methods. It is used, e.g., to compress the input data into a lower-dimensional representation, which is later employed in other machine learning algorithms. In the area of quantum computing application methods, quantum machine learning is also developing dynamically, with the concept of a quantum autoencoder also present. In the article, we discuss a variant of building a quantum autoencoder based on a quantum convolutional network. The proposed autoencoder is characterized by an architecture based on adjacent quantum gates, which is especially important for the near-future noisy intermediate-scale hardware implementation of such a type of quantum circuits. The proposed circuits are built with an elementary set of gates, i.e., controlled negation gates and rotation gates. The described architecture of a quantum autoencoder properly uses the quantum phenomenon called quantum exponential capacitys, i.e., a linear number of qubits which allow encoding an exponential amount of classical information. In our case, for n qubit, it is possible to encode a classical gray coloured image with dimensionality $2^n \times 2^n$. The conducted numerical experiments on a set image of handwritten characters and letters show that, for a small number of parameters, the quantum autoencoder offers a reconstruction quality comparable to the currently used classical autoencoders. An important assumption is the omission of additional dimensionality reduction techniques, e.g., PCA, for the preparation of classical data. In the discussed experiment, the data after reconstruction can be read directly from the quantum register, through a series of quantum register measurements.

Keywords: autoencoder, quantum autoencoder, convolution quantum circuit.

1. Introduction

The contemporary development of quantum computing technology (Nielsen and Chuang, 2010) in the theoretical and experimental fields (Ritter, 2019; Preskill, 2018; Sivarajah *et al*., 2020; Brooks, 2019) opens new perspectives for the development of quantum machine learning (QML), which can revolutionize data processing and data representation. One promising area of research is the application of quantum autoencoders (QAEs), which serve as quantum analogues of classical autoencoders (AEs), for efficient data compression and the extraction of meaningful features. Neural autoencoders have been widely used in classical machine learning for dimensionality reduction, denoising, and generative modeling tasks. Although classical autoencoders have demonstrated significant progress in image processing, their quantum counterparts exploit the distinctive properties of quantum mechanics, including superposition and entanglement. These may enable more efficient encoding of information. Nevertheless, the effectiveness of QAEs compared to that of classical methods remains an open research question, particularly in relation to image-based datasets.

This work proposes a new approach to the evaluation of quantum and classical autoencoders by testing them on a custom dataset of handwritten Polish characters. It consists of 64×64 pixel grayscale images, prepared in a process similar to the MNIST dataset, with modifications tailored to the Polish alphabet. Due to the lack of publicly available datasets containing handwritten Polish letters, ours provides a valuable contribution to the development of classical and quantum machine learning by enabling a reliable assessment of various machine learning methods.

*Corresponding author

The main objectives of this study are as follows:

- to design and implement a QAE architecture capable of compressing and reconstructing images encoded in a quantum system;
- to create a benchmark dataset of handwritten characters from the Polish alphabet;
- to develop and implement classical autoencoders as a baseline for comparison with quantum models;
- to perform an experimental analysis and compare both approaches regarding image reconstruction quality.

The quantum and classical models are evaluated based on reconstruction quality, dimensionality reduction, and computational efficiency. Due to the limitations of current quantum devices, including a small number of qubits and high noise levels, special attention is given to optimizing the quantum circuit used in the QAE. The results provide valuable insights into quantum autoencoders' current capabilities and limitations. They also highlight the potential benefits of quantum methods compared to traditional neural networks. This research contributes to quantum machine learning by demonstrating a practical application of quantum autoencoders in image processing and comparing their performance with that of classical approaches. In addition, creating a new benchmark dataset of handwritten Polish characters introduces new research possibilities in image processing, optical character recognition (OCR), and machine learning methods adapted to the Polish language. The findings of this study are also crucial for developing QML in image analysis and quantum data compression. They represent a step toward practical quantum computing applications in real-world data processing tasks.

Section 2 reviews the existing literature on quantum autoencoders. Section 3 introduces the notation used in the article. Section 3.2 presents the image dataset with Polish characters prepared for the experiments. Section 4 describes the architecture of a quantum autoencoder which is based on a quantum convolution network. Section 5 describes the experimental setup, evaluation metrics and computational limitations, and also provides simulation results, analysis, and comparison of classical and quantum approaches to the reconstruction of handwritten digits and letters. Finally, Section 6 summarizes the key findings, presents the main conclusions, and outlines directions for future research on quantum autoencoders.

2. Related works

The origins of autoencoders can be traced back to studies from the late 1980s and early 1990s, in which neural networks began to be viewed as tools for data compression and representation learning. Foundational contributions such as those by Cottrell *et al*. (1987), Baldi and Hornik (1989) or Kramer (1991; 1992) demonstrated that unsupervised networks could reconstruct input data by encoding it internally into a lower-dimensional latent space. Significant progress was made with the emergence of deep autoencoder architectures. A breakthrough work by Hinton and Salakhutdinov (2006) introduced a deep autoencoder pre-trained using restricted Boltzmann machines (RBMs), followed by fine-tuning with backpropagation. Subsequent years saw the development of various architectural variants, including denoising autoencoders (Vincent *et al.*, 2008; 2010), convolutional autoencoders (Masci *et al*., 2011), variational autoencoders (VAEs) introducing probabilistic latent encoding (Kingma and Welling, 2014), adversarial autoencoders (AAEs) combining reconstruction with latent space regularization via a discriminator (Makhzani *et al*., 2015), autoencoders for learning disentangled representations (Le *et al*., 2012; Higgins *et al*., 2017), as well as models based on attention mechanisms and transformer architectures (Devlin *et al*., 2018; He *et al*., 2022).

This dynamic development has led autoencoders to become, in recent years, one of the key tools in many domains, including image compression and reconstruction (Al-Khafaji and Ramaha, 2025), anomaly detection (Aslam *et al*., 2024; Shang *et al*., 2024), image processing, analysis and generative modeling or style transfer (Luhman and Luhman, 2023; Liu *et al*., 2021).

While modern deep neural autoencoders successfully address complex engineering problems, their quantum counterparts are still in the early stages of development. They are subject to significant technological constraints, such as a limited number of available qubits, shallow circuit depth, and high sensitivity to noise. For this reason, a direct comparison between quantum autoencoders and the most advanced classical architectures would not be methodologically justified. Therefore, this study adopts the reconstruction of grayscale images of handwritten letters and digits as a reference task. This simplified yet precisely defined experimental setting enables a fair and reliable comparison of quantum models with well-established classical solutions.

The reconstruction of handwritten letters and digits using neural autoencoders has been extensively studied in the literature. For instance, Janjua and Patankar (2024) proposed denoising convolutional autoencoders trained and tested on the MNIST and Fashion-MNIST datasets, in the context of image retrieval for visually similar samples. Classical autoencoders have also been applied to dimensionality reduction tasks involving MNIST, CIFAR-10 and Fashion-MNIST, often demonstrating superiority over traditional techniques such as PCA or

Table 1. Some symbols, notation, sets and functions used in the paper.

Notation	Description
X	set of classical data (letters and digits)
X_j, Y_j	single image of j-th char and output image from decoder
N_I	number of images
$\mathcal{E}, \mathcal{D}$	encoder and decoder
θ	trainable set of parameters
N, L	total qubits in quantum register N and L qubits used in latent/compressed space
N_Q	number of qubits using to encode an image
$\mathcal{L}$	loss function
$\mathcal{F}$	fidelity of two quantum states
i, j, k	integer numbers usually used as indices
$\mathbb{R}$, $\mathbb{C}$, $\mathbb{N}$	sets of real, complex, and integer numbers
$\otimes$	Kronecker product of matrices or vectors
$\oplus$	direct sum of matrices and spaces
$\mathrm{tr}\,(-)$	trace of operator/matrix
$\mathrm{tr_S}\,(-)$	partial trace of operator/matrix for subsystem S
$\langle -\vert -\rangle$	inner product
$\sigma(A)$	spectrum of matrix A (counted with multiplicities)
$\partial E(\mathbb{C}^d)$	set of pure states on $\mathbb{C}^d$
$1 \ldots n$	sequence of $1, 2, 3, \ldots, n$

Isomap in capturing nonlinear data structures (Fournier and Aloise, 2019). In the context of anomaly detection, Neloy and Turgeon (2024) conducted a comparative analysis of 11 autoencoder architectures using the MNIST and Fashion-MNIST datasets, highlighting differences in reconstruction quality, computational cost, and anomaly sensitivity. An example of autoencoder-based classification of handwritten digits can be found in the work of Loey *et al*. (2017), where a deep autoencoder was applied to recognize handwritten numerals.

The dynamic development of quantum computing methods (Nielsen and Chuang, 2010), as well as the area of quantum machine learning (Ciliberto *et al*., 2018; Zeguendry *et al*., 2023; Huang *et al*., 2021), has also led to increased interest in the area of application and properties of autoencoders. The concept of an autoencoder (termed AE) was relatively quickly transferred to the field of quantum computing methods. In the works of Romero *et al*. (2017) and Wang *et al*. (2024) the basic definition of a quantum autoencoder (QAE) was introduced as a solution used for the compression of quantum data. The compression ratio was also discussed by Ma *et al*. (2023). The problem of data coding using a QAE is also widely discussed (Bravo-Prieto, 2021). One of the potential applications of autoencoders is denoising, e.g., images; a discussion on this issue in the context of quantum methods was carried out by Cao and Wang (2021). Examples of real industrial applications were presented by Mangini *et al*. (2022). Basic results regarding the definition of quantum circuits that form autoencoders were given by Wu *et al*. (2024).

It should also be added that well-known classical datasets, such as MNIST, are also already being used in the context of quantum computing methods, e.g., Kurowski *et al*. (2021) employ the restricted Boltzmann machine based on quantum annealing for a classification task while Slysz *et al*. (2023) explore the application of a quantum support machine for the classification of digits.

3. Preliminaries

Before starting the presentation of a quantum autoencoder based on a quantum convolutional circuit, a selection of the most important abbreviation symbols and acronyms used in this paper is presented in Table 1.

In this work we write a quantum state in a Dirac notation, i.e., $|\psi\rangle$, which is a d-dimensional column vector, while $\langle\psi|$ is a row vector resulting from the transposition of the vector $|\psi\rangle$. In general, these vectors can be described by complex numbers, therefore $|\psi\rangle \in \mathbb{C}^d$, where $\mathbb{C}^d$ is the Euclidean d-dimensional space. In our case we will use the notion of a qubit, so the vector $|\psi\rangle$ representing a quantum state of N qubits also satisfies the following condition:

$$|\psi\rangle = \sum_{i=0}^{2^N-1} \alpha_i |i\rangle \quad \text{and} \quad \sum_{i=0}^{2^N-1} |\alpha_i|^2 = 1, \tag{1}$$

where $\{|i\rangle\}$ are orthonormal basis vectors and α_i are the so-called probability amplitudes. As will be shown further (Section 3.3), the quantum state in such a notation

allows efficient recording of information about a classical two-dimensional image.

Let us add that we also use the concept of a density matrix, which is created by the matrix product of the column and row vectors:

$$\rho = |\psi\rangle\langle\psi|. \tag{2}$$

Further basic information about the definition of the quantum information notion can be found in many currently available textbooks on quantum computing (e.g., Nielsen and Chuang, 2010).

It should be also added that, in the case of the spectrum of the operator ρ representing the quantum state as a density matrix,

$$\rho = \sum_{i=1}^{k} \lambda_i |\psi_i\rangle\langle\psi_i|, \tag{3}$$

where the set of vectors $\{\psi_i\}$ represents the orthonormal basis while λ_i are eigenvalues. The eigenvalues $\{\lambda_i\}$ together with the set vectors $\{\psi_i\}$ will be presented in descending order with respect to the values of λ_i, i.e.,

$$\sigma(\rho) = \lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_{k-1} \geq \lambda_k. \tag{4}$$

To evaluate the quality of the quantum autoencoder and to tune its parameters, the fidelity measure will be applied, which, if the quantum states are described by a density matrix, is expressed as follows:

$$\mathcal{F}(\rho,\sigma) = \left(\mathrm{tr}\left(\sqrt{\sqrt{\rho}\sigma\sqrt{\rho}}\right)\right)^2. \tag{5}$$

Since pure quantum states will be used in the evaluation of a quantum circuit, a simplified form of the fidelity measure can be employed:

$$\mathcal{F}(\rho,\sigma) = |\langle\psi_\rho|\psi_\sigma\rangle|^2. \tag{6}$$

3.1. Autoencoder. A classical autoencoder is a type of artificial neural network designed to learn efficient representations of input data in an unsupervised manner. It is composed of two fundamental components: an encoder $\mathcal{E}$, which transforms the input data X_j into a lower-dimensional space known as the latent/compressed space, and a decoder $\mathcal{D}$, which reconstructs the original input from this compressed representation (Fig. 1). The training process involves minimizing the reconstruction error between the input and the output image $Y_j = \mathcal{D}(\mathcal{E}(X_j))$, typically using the mean squared error (MSE) as the loss function. An important property of the autoencoder is high quality of an output image reproduction, with the smallest possible compressed/laten space dimensionality.

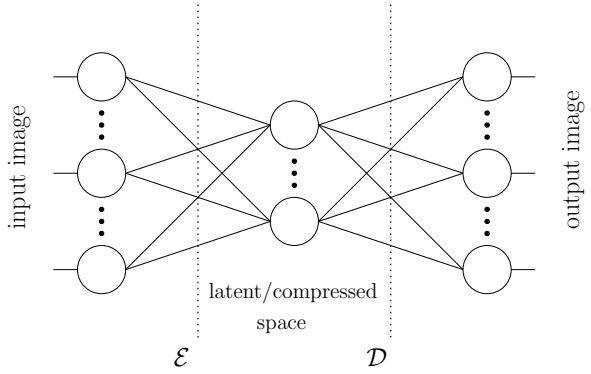

Fig. 1. General idea of a classical autoencoder, where an input image X_j is given as input, and then the neural network $\mathcal{E}$ representing the encoder converts the input image into a representation of lower dimensionality. The task of the second part, $\mathcal{D}$, of the autoencoder is to convert the compressed space back to the same data dimensionality as the output data.

Modern autoencoders are typically based on deep neural network architectures, enabling them to capture complex and non-linear dependencies within the data. As a result, they have become one of the key tools for unsupervised representation learning, allowing the extraction of hidden structures without the need for labeled data. Although the learning process is based on optimizing a cost function, the model receives no external information about the meaning of the input, since the target output is identical to the input itself. This setup allows the autoencoder not only to perform data compression but also learn a new, informative representation space (latent space). Within this latent space, important semantic features, hidden class structures, and other data regularities may be revealed, even though the model is not explicitly guided to recognize them.

An essential element of the autoencoder architecture is the so-called bottleneck layer. It has a reduced number of neurons, forcing the model to identify and retain only the most relevant features necessary for faithful reconstruction. The dimensional constraint imposed by this layer leads to the formation of a compact and informative latent representation.

3.2. Data set. The quantum autoencoder proposed in this work as well as the reference classical convolutional autoencoder were trained and tested on a specially prepared dataset named PolLettDS (Sawerwain and Kowal, 2025a). This dataset was developed as part of a project aimed at collecting handwritten digits as well as lowercase and uppercase letters of the Polish alphabet. Although numerous publicly available datasets contain handwritten text, such as MNIST for

Table 2. Summary of the PolLettDS dataset (files available at `https://github.com/qMSUZ/PolLettDS`).

Total number of images	4160
Number of participants	52
Character types	digits, lowercase and uppercase Polish letters
No. of digits	520 (10 per participant)
No. of lowercase letters	1820 (35 per participant)
No. of uppercase letters	1820 (35 per participant)
Data sources	paper and electronic
Paper/electronic data ratio	50%/50%
Image resolution	64×64 pixels
Image format	8-bit grayscale
Color scheme	black background, light characters
File format	2D raw raster images
License and availability	MIT license, GitHub (Sawerwain and Kowal, 2025a)

digits or EMNIST for Latin letters, there is only one dataset (Tokovarov *et al*., 2020) specifically tailored to Polish characters. This dataset reflects the whole Polish alphabet, including diacritical letters such as "ą", "ć", "ł", and "ź". Language-specific datasets are crucial for advancing optical character recognition (OCR) systems, particularly for languages with unique character sets. Therefore, it was decided to create a new dataset, with particular emphasis on higher resolution because the dataset of Tokovarov *et al*. (2020) contains characters sized 32×32. Moreover, the characters in PolLettDS were obtained not only by scanning paper documents but also through digital acquisition.

The PolLettDS dataset was created in Poland at the University of Zielona Góra, where 52 volunteers completed forms containing fields for eighty characters (10 digits as well as 35 lowercase and 35 uppercase letters). The forms were available in two versions: a digital, filled out using a digital pen on a graphics tablet, and a paper one, which was scanned after being completed. The data from the electronic forms were saved as grayscale images, with light characters on a dark background. The scanned paper forms underwent preprocessing, including grayscale conversion and color inversion, so that their format matched that of the electronic version (dark background, light characters). Subsequently, all forms, regardless of the data source, were automatically cropped into 64×64 pixel images. As a result, 4160 character images in raster format were obtained. Sample images are shown in Fig. 2. For the purpose of training the autoencoders, the dataset was randomly split into training, validation, and test subsets in a 7:2:1 ratio. The images were also normalized so that pixel values ranged from 0 to 1.

A detailed summary of the dataset is provided in Table 2. The data is publicly available in a GitHub repository and can be used for research purposes.

3.3. Data encoding. Data representation, especially of classical data in a quantum register, is an important issue that can affect the correct use of the features of a quantum computational model, i.e., the superposition and quantum operations. A single image X_j for the discussed form of a quantum autoencoder is encoded using the so-called amplitude coding approach (Weigold *et al*., 2021):

$$X_j \longmapsto |\psi_j\rangle = \sum_{i=0}^{2^N-1} \alpha_i |i\rangle, \tag{7}$$

where probability amplitudes α_i describe information about the luminance level for a given pixel and base state $|i\rangle$ represents coordinates of a given pixel. In other words, the two-dimensional X_j image with the gray level luminance pixel is normalized and reorganized as the column vector, which can be directly interpreted as the pure quantum state $|\psi_j\rangle$.

The advantage of using probability-amplitude coding is the possibility of employing the quantum superposition property. For the PolLettDS dataset and a single gray image with a resolution of 64×64 pixels, 4096 bytes of information are needed per image. The corresponding quantum register that will represent the same image requires the use of only 12 qubits, which means that the number of qubits is exponentially smaller in relation to classical information.

However, there is one example where 12 qubits will unfortunately not be sufficient to correctly record image information, and this is the case of an empty image containing pixels with brightness described by zero. Here, one extra qubit is needed, which indicates that we have this one special case. In general, the number of qubits needed in the case of encoding an image of resolution $p \times p$ is

$$N_Q = \log_2(p^2) + 1. \tag{8}$$

This means that a given classical image X_j from

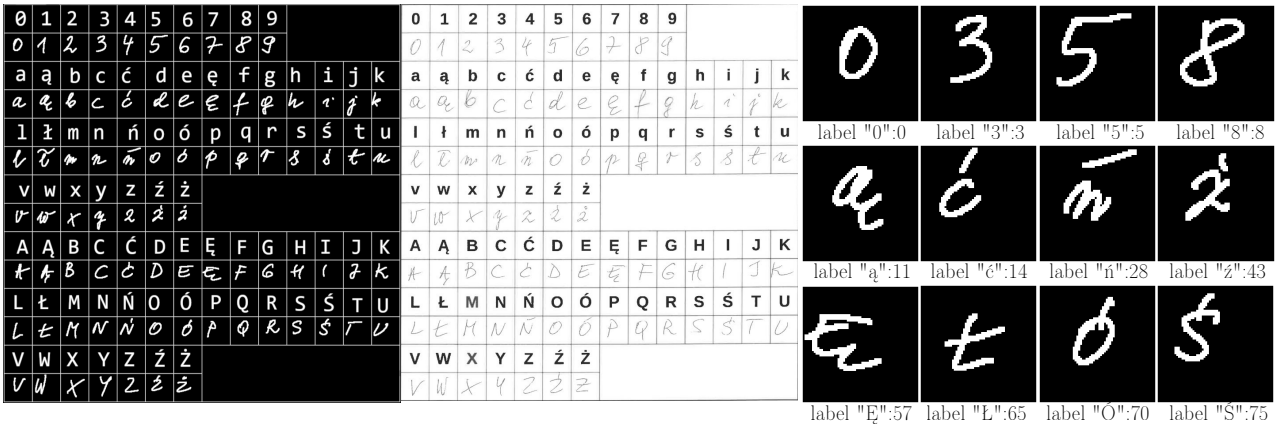

Fig. 2. Sample handwritten letters and digits from the dataset, collected from volunteers using both paper-based and tablet-based digital forms.

the PolLettDS can be encoded using a significantly lower amount of qubits.

4. Quantum autoencoder defined by a convolutional circuit

In this part of the article, we present the construction of a quantum autoencoder using a quantum convolutional network. Before giving the construction, we also discuss the theoretical foundations of the existence of quantum autoencoders as unitary operations. Then, we show that the construction we propose is subject to the upper bound theorem on the quality of data reconstruction by the autoencoder. The last part of this section covers specification of the form of the quantum circuit implementing the autoencoder, together with the definition of the loss function used during the process of optimizing the autoencoder operation.

4.1. Quantum autoencoder definition. Based on the works of Romero *et al*. (2017), Wu *et al*. (2024) as well as Du and Tao (2025), we denote by $\mathcal{E}(\theta)$ the operation of encoding data with parameters θ (these are naturally subject to the training procedure), for the input state $|\psi\rangle$: $\mathcal{E}(\theta)(|\psi\rangle) = |\mathrm{aux}\rangle \otimes |\phi\rangle$, where $|\mathrm{aux}\rangle$ denotes the state containing information that will be omitted in the state decoding process: we only use the state $|\phi\rangle$ which represents the compressed input state in a smaller subspace, also known as a compressed or latent space. An operation of this type can be represented by a unitary one and the input vector is transformed into the Kroneker product of two consecutive quantum registers. This product also means that the input register can be entangled, and the encoder's task will be to disentangle the register and express the input state $|\psi\rangle$ as a product of the states $|\mathrm{aux}\rangle$ and $|\phi\rangle$.

The task of the decoder ($\mathcal{D}(\theta)$, also expressed as a unitary operation) is to restore the input state: $\mathcal{D}(\theta)(|\mathrm{aux}\rangle \otimes |\phi\rangle) = |\psi\rangle$.

From the basic properties of unitary operations and Stone's theorem (the theorem and some remarks on unitary evolution are given in Appendix), the following theorem is derived which describes an ideal quantum autoencoder based on unitary operators, where by $\xrightarrow{\mathcal{D}}$ we denote application of the decoder operation.

Theorem 1. (Existence of a quantum autoencoder) *If $|\psi\rangle$ represents the input state and the unitary operators $\mathcal{E}(\theta)$, $\mathcal{D}(\theta)$ represent the encoding and decoding operations, then a quantum autoencoder exists and is expressed as*

$$\begin{aligned} \mathcal{E}(\theta)(|\psi\rangle) &= |\mathrm{aux}\rangle \otimes |\phi\rangle \\ &\xrightarrow{\mathcal{D}} \mathcal{D}(\theta)(|\mathrm{aux}\rangle \otimes |\phi\rangle) \\ &= |\psi\rangle. \end{aligned} \tag{9}$$

Proof. The proof of the existence of a quantum autoencoder is directly related to the basic property of unitary operations on the quantum register. From Stone's theorem (Theorem A1), for two given quantum states $|\psi_1\rangle$ and $|\psi_2\rangle$, it is known that there always exists unitary operation U where $U|\psi_1\rangle = |\psi_2\rangle$. Additionally, for any unitary U, it is always to possible to create an inverse operation by using hermitian adjoint $U^\dagger$: $U^\dagger|\psi_2\rangle = |\psi_1\rangle$, naturally; $UU^\dagger = \mathbb{I}$ and the symbol $\mathbb{I}$ represents an identity operator.

Taking into account the mentioned basic properties, the quantum autoencoder can be expressed as

$$\begin{aligned} \mathcal{E}(\theta)(|\psi\rangle) &= |\mathrm{aux}\rangle \otimes |\phi\rangle \\ &\xrightarrow{\mathcal{D}} \mathcal{D}(\theta)(|\mathrm{aux}\rangle \otimes |\phi\rangle) \\ &= \mathcal{D}(\theta)\mathcal{E}(\theta)(|\psi\rangle) \\ &= \mathcal{E}(\theta)^\dagger \mathcal{E}(\theta)(|\psi\rangle) \\ &= |\psi\rangle, \end{aligned} \tag{10}$$

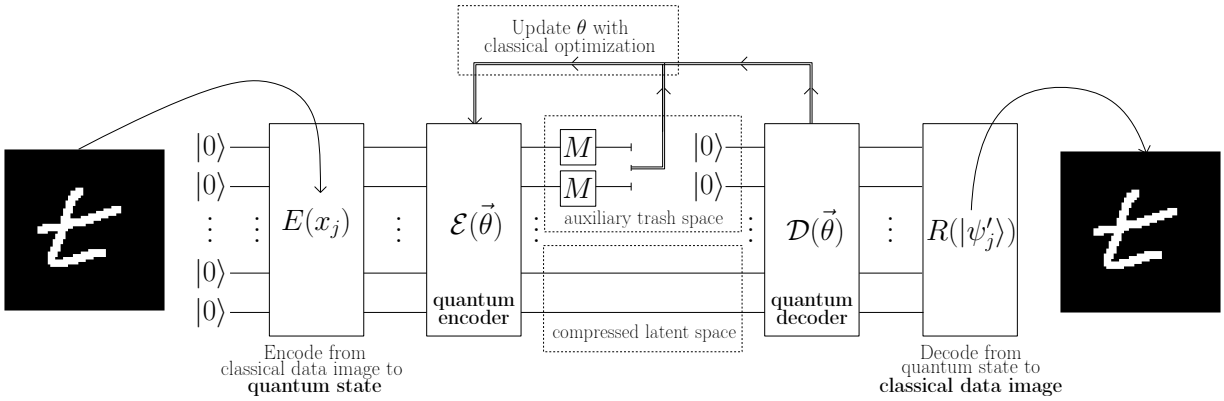

Fig. 3. Structure of a quantum autoencoder. Similarly to the classical version, there is an encoder part $\mathcal{E}$ and a decoder part $\mathcal{D}$. We will also use the notation $n-l-n$ to indicate that the input image is encoded using exactly n qubits, the latent space needs l qubits, and the output image is represented again by the n qubits.

where $\mathcal{D}(\theta)=\mathcal{E}(\theta)^\dagger$, which concludes the proof. ■

The existence of a quantum autoencoder means that the described scheme of the classical autoencoder presented in Section 3.1 can be also transferred to the case of a quantum autoencoder, which is shown in Fig. 3. The presented general scheme uses the operations $\mathcal{E}(\theta)$ and $\mathcal{D}(\theta)$, which function on the entire quantum register of n qubits. The compressed or laten space is restricted to l qubits.

It should be noted that the input state $|\psi\rangle$ should be regarded as a system of two subsystems, therefore we are going to write this state in the form of a density matrix $\rho_\psi=|\psi\rangle\langle\psi|$ with the following decomposition: $\rho_{AB}=\sum_i \lambda_i|\Psi_j\rangle\langle\Psi_j|$.

The intermediate state that we get after executing the encoder operation $\mathcal{E}(\theta)$ is obtained using a partial trace, i.e., the state $|\psi^A\rangle$, written as $\rho_A=|\psi^A\rangle\langle\psi^A|$, and after executing the partial trace:

$$\rho_A=\mathrm{tr}_\mathrm{B}\left(\mathcal{E}(\theta)\rho_{AB}\mathcal{E}(\theta)^\dagger\right), \tag{11}$$

where $\mathcal{E}(\theta)$ is in this case a unitary operator representing the encoder. The subsystem B in this case naturally represents the latent space. Regarding the decoder operation, the state ρ_A is, as shown in Fig. 3, initialized to the state $|0\rangle$, and we use $n-l$ qubits in this case.

Remark 1. Anticipating further discussion, we recall that in the description of the quantum autoencoder in Section 4.1 and its training process a measurement operation is used, which represents a non-unitary computational step. However, the goal of the training procedure is naturally to approximate the behavior of the autoencoder described by Theorem 1, in the sense that the specified loss function estimating the difference between the input and output states takes on the lowest possible value.

4.2. Upper bound of fidelity for the reconstructed state. Cao and Wang (2021) provide a theorem on autoencoder maximal fidelity, and the autoencoder proposed in this work also fulfills the following theorem.

Theorem 2. (Upper bound of fidelity for the reconstructed state) *If the input state ρ_{X_j} has the spectrum $\sigma(\rho_{X_j})=\sum_{i=1}^k\lambda_i|\psi_i\rangle\langle\psi_i|$, where the vectors $\{\psi_i\}$ stand for orthonormal base and the eigenvalues λ_i are ordered to decrease $\{\lambda_i\}_{i=1}^k$, then the fidelity of the autoencoder $\mathcal{F}$, where the encoder is represented by $\mathcal{E}$ and the decoder by $\mathcal{D}$ (we assume that $\mathcal{D}=\mathcal{E}^\dagger$), is trained for parameters θ), has the upper bound $\mathcal{F}\le\sum_{j=1}^{d_A}\lambda_j$.*

Proof. First, we assume that the obtained reconstructed state ρ_{Y_j} has the spectrum expressed as

$$\rho_{Y_j}=\sum_{i=1}^{d_A}\kappa_i|\phi_i\rangle\langle\phi_i|, \tag{12}$$

where spectrum $\{\kappa_i\}_{i=1}^{d_A}$ is ordered as a decreasing sequence.

Taking into account positive operator-valued measure set $\{M_l=|\Phi_l\rangle\langle\Phi_l|\}_{l=1}^{d_Ad_B}$, where Φ_l are orthonormal vectors, we obtain the following relation for upper bound fidelity for input state (ρ_{X_j}) and reconstructed state ρ_{Y_j}:

$$\begin{aligned}\mathcal{F}(\rho_{X_j},\rho_{Y_j})&\le\left(\sum_{i=1}^{d_Ad_B}\sqrt{\mathrm{tr}\left(\rho_{X_j}M_i\right)\mathrm{tr}\left(\rho_{Y_j}M_i\right)}\right)^2\\&=\left(\sum_{i=1}^{k}\sqrt{\lambda_i\langle\Phi_i|\rho_{Y_j}|\Phi_i\rangle}\right)^2\end{aligned}$$

$$\leq \left(\sum_{i=1}^{d_A}\sqrt{\lambda_i\kappa_i}\right)^2 \leq \left(\sum_{i=1}^{d_A}\lambda_i\right)\left(\sum_{i=1}^{d_A}\kappa_i\right) = \sum_{i=1}^{d_A}\lambda_i. \quad (13)$$

According to Cao and Wang (2021) the first inequality reuses the fact that fidelity cannot be larger than fidelity after measurement (Nielsen and Chuang, 2010). Therefore, the autoencoder also cannot increase the fidelity between two states. The second inequality is attained by rearrangement and the fact that we use the decreasing spectrum of ρ_{Y_j}, while the last transformation is obtained by the application of the Cauchy–Schwarz inequality. ■

4.3. Quantum convolutional circuit as a quantum encoder. The quantum circuit which is used to implement the quantum autoencoder uses a quantum convolutional neural network (QCNN); in general, the construction of the quantum convolutional network is presented by Cong *et al*. (2019). The circuit uses a set of parameters $\vec{\theta}$, which are trained with the classical optimization algorithm; in our case it will be the COBYLA method (Powell, 1994; 2007). A characteristic feature of a quantum convolutional network, is that individual layers include a smaller and smaller number of qubits, which corresponds to a classical convolutional network where the dimensionality of the network decreases. In our case, we will strive to reduce the dimensionality by omitting selected qubits, e.g., they will assume previously determined quantum states. It should also be noted that the operation of freely expanding the structure in the case of quantum networks is not possible, and similarly the operation of narrowing the structure of a convolutional network is usually carried out in the case of a quantum circuit by means of the operation of measuring or omitting specific qubits during the creation of individual layers, although they remain present in the circuit. This naturally means that the number of qubits does not change throughout the processing of a given circuit.

Figure 4 presents a schematic diagram of an autoencoder based on a quantum convolutional network. The unitary operation implementing the first part of the autoencoder, i.e., the encoder with n layers, is therefore a sequence of unitary operators QC_k, QP_k belonging to layer k together with parameters referring to the convolution $\theta_{k,c}$ and polling $\theta_{k,p}$ subcircuits:

$$\mathcal{E}(\theta) = QC_1(\theta_1^C)\cdot QP_1(\theta_1^P)\cdot \ldots \cdot QC_k(\theta_k^C)\cdot QP_k(\theta_k^P). \quad (14)$$

Generally, a convolution cirucit (which act on N_Q qubits) is denoted as follows:

$$QC_n(\theta_{n1}) = \left(\prod_{i=0}^{N_Q} R_y^{(i)}(\theta_{n_1}^k)R_z^{(i)}(\theta_{n_1}^l)\right) \times \left(\prod_{i=0}^{N_Q-1} CR_x^{(i,i+1)}(\theta_{n_1}^m)\right), \quad (15)$$

where $k,l,m \in \{1\ldots N_Q\}$, $R(i)_z(\theta_{n_1}^k)$ means that a given gate is applied to the i-th qubit, while $CR_x^{(i,i+1)}(\theta_{n_1}^m)$ represents a controlled rotation gate, where i is the controlling qubit and $i+1$ is the controlled qubit.

The definition of the polling layer can be expressed as follows:

$$QP_n(\theta_{n2}) = \prod_{i=0}^{N_Q-1} CR_z^{(i,i+1)}(\theta_{n_1}^k)X^{(i)}CR_z^{(i,i+1)}(\theta_{n_1}^l), \quad (16)$$

where the notation remains the same as in the convolutional layers.

The decoder operation can be rewritten in the opposite way; the last gates used in the encoder stand for the first gate for decoder, and the all operators on the given layer are used with the application of the Hermitian adjoint:

$$\mathcal{D}(\theta) = QP_k^\dagger(\theta_k^P)\cdot QC_k^\dagger(\theta_k^C)\cdot \ldots \cdot QP_1^\dagger(\theta_1^P)\cdot QC_1^\dagger(\theta_1^C). \quad (17)$$

Thus, preserving the values of the set of trained parameters θ thanks to the change in the order of application of operators and the Hermitian adjoint operator, the decoder works inversely to the encoder, in accordance with the definition of the Hermitian adjoint for unitary operators. In the case of single-rotation gates, only the negation of the angle rotation value is enough to inverse the operation of a given gate.

4.4. Loss function. The autoencoder training process, in the case of the proposed solution, will naturally consist of the appropriate selection of θ values, which are common to the encoder and the decoder. Since we use quantum states, the evaluation of reconstruction quality can also be realized with the fidelity measure, for a specific j-th input state $|X_j\rangle$ and output state $|Y_j\rangle$:

$$\mathcal{L}(X_j,Y_j) = 1 - \mathcal{F}(|X_j\rangle,|Y_j\rangle). \quad (18)$$

Since $\mathcal{F}$ takes values $\langle 0,1\rangle$, where unity means that two quantum states in the sense of the fidelity measure are identical, for the correct interpretation of the loss function we subtract its value from unity.

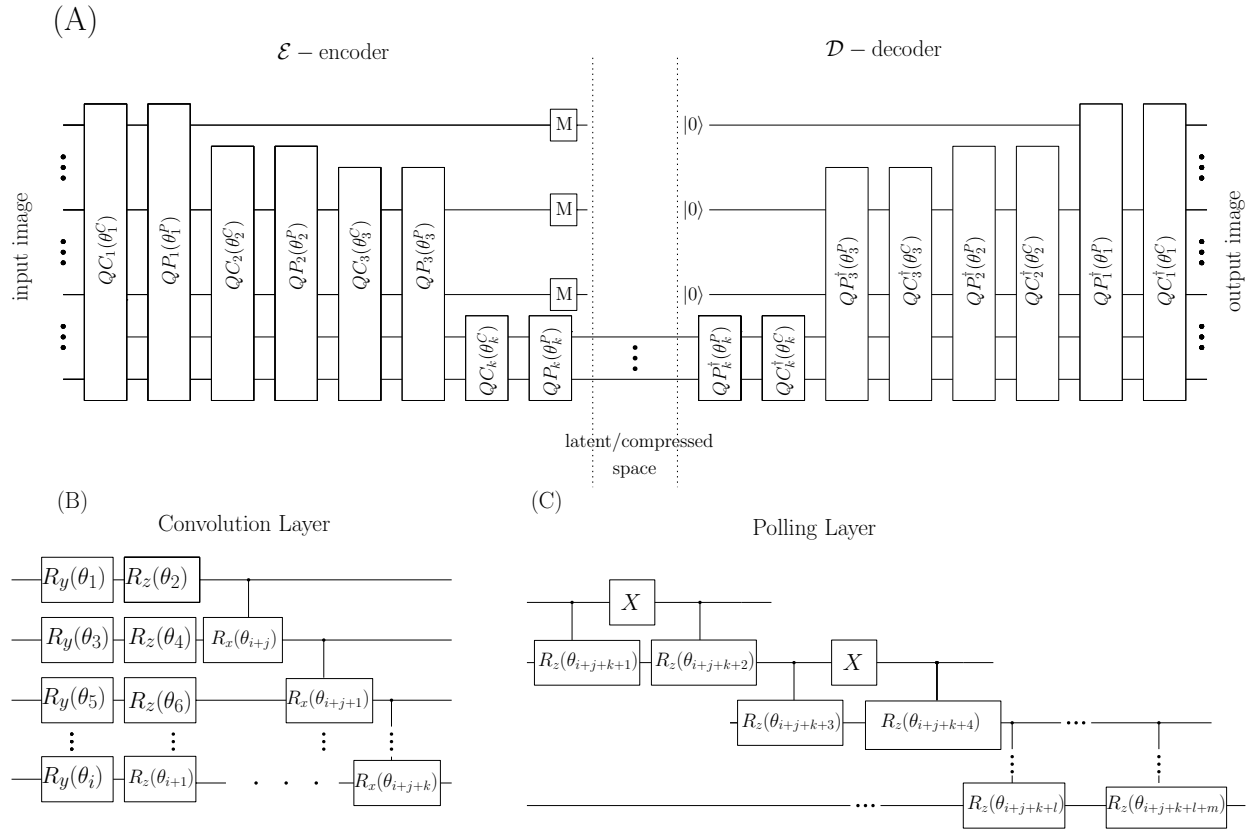

Fig. 4. General structure of the autoencoder circuit using the QCNN consists of a series of convolution (QC) and polling (QP) operators (a). The symbol † denotes the Hermitian conjugate for a specific operator representing the QC and QP layers. The construction of a single convolution layer (b) is formed using the rotation gates Y and Z, while the rotation gate X is a controlled gate. Part (c) presents the general construction of the combining layer. An important assumption is that all gates are adjacent.

When we perform training on the entire data set, the loss function is as follows:

$$\mathcal{L}(X,Y) = \frac{1}{N_I}\sum_{j=0}^{N_I-1} 1 - \mathcal{F}(|X_j\rangle, |Y_j\rangle). \tag{19}$$

The minimization of the value of the function $\mathcal{L}$ occurs naturally by selecting the values of the parameters θ:

$$\min_{\theta} \quad \mathcal{L}(X,Y). \tag{20}$$

In the numerical experiment (Section 5), the minimization is performed with the COBYLA method.

4.5. Number of parameters. The number of parameters used on the encoder $\mathcal{E}$ will be related to the total number of qubits N employed to encode images from the set X and the size of the latent space L. The number of parameters for the encoder $\mathcal{E}$ is therefore determined by the following relation:

$$N_\theta = \sum_{k=(N-(N-L))}^{N} \left(3_c^k + 2_p^k\right), \tag{21}$$

where 3_C^k stands for the number of parametrized gates in the k-th convolutional layer and 2_P^k, analogously, for the same in the k-th polling layer. For the case considered in the article, $N = 12$, if the laten space is equal to 10 qubits ($L = 10$), then a total number of parameters is equal to $156 = 57 + 52 + 47$ for the encoder $\mathcal{E}$ (naturally, the decoder $\mathcal{D}$ shares the same parameters).

More of the technical details of the construction and simulation of the quantum autoencoder can be found in the source code repository, where there is an example simulation of the autoencoder we proposed (Sawerwain and Kowal, 2025b).

5. Numerical simulations and experiments

The proposed quantum convolutional autoencoder was tested on the task of reconstructing letters and digits from the PolLettDS dataset. A detailed description of the model architecture is provided in Section 4. The numerical experiments used a quantum autoencoder composed of a three-layer encoder and a three-layer decoder, with alternating convolutional modules and fusion layers (Fig. 4). Input images were represented

Table 3. Architecture of Classical Autoencoder I.

Layer	Type	Output	Details
1	Input	64×64×1	input
2	Conv2D	32×32×1	kernel 3×3, stride 2, ReLU
3	Conv2D	16×16×4	kernel 3×3, stride 2, ReLU
4	Conv2D$^\top$	32×32×2	kernel 3×3, stride 2, ReLU
5	Conv2D$^\top$	64×64×1	kernel 3×3, stride 2, ReLU
6	Conv2D	64×64×1	kernel 3×3, stride 1, sigmoid

Table 4. Architecture of Classical Autoencoder II.

Layer	Type	Output	Details
1	Input	64×64×1	input
2	Conv2D	32×32×4	kernel 3×3, stride 2, ReLU
3	MaxPool2D	16×16×4	pool 2×2, stride 2
4	Conv2D	8×8×64	kernel 3×3, stride 2, ReLU
5	MaxPool2D	4×4×64	pool 2×2, stride 2
6	Conv2D	2×2×1024	kernel 3×3, stride 2, ReLU
7	MaxPool2D	1×1×1024	pool 2×2, stride 2
8	Conv2D$^\top$	4×4×64	kernel 4×4, stride 1, ReLU
9	Conv2D$^\top$	8×8×16	kernel 3×3, stride 2, ReLU
10	Conv2D$^\top$	16×16×4	kernel 3×3, stride 2, ReLU
11	Conv2D$^\top$	32×32×1	kernel 3×3, stride 2, ReLU
12	Conv2D$^\top$	64×64×1	kernel 3×3, stride 2, ReLU
13	Conv2D	64×64×1	kernel 3×3, stride 1, sigmoid

using 12 qubits (corresponding to a state space of size 2^{12}), while the latent space was described using 10 qubits (size 2^{10}). The model contained a total of 156 parameters, optimized using the COBYLA algorithm. The maximum number of training epochs was set to 2048. The training process aimed to minimize the loss function defined in Eqn. (19). The training set was limited to only five characters due to the computational time needed for the optimization routine. The average learning time was 83 seconds for a modern desktop workstation with an AMD Ryzen 9 7950X processor and an NVIDIA RTX 6000 ADA graphics card. Computations were performed on the WSL 2 environment for Windows 11, where the NVIDIA CUDA Q v0.11.0 package was used. This is a significantly longer learning time, as only five characters were used as a learning set. In the case of classic autoencoders, the learning time averaged 72 seconds for a training set that contains 2908 images.

The performance of the proposed quantum autoencoder was compared with that of classical convolutional autoencoders. The aim was to assess whether the quantum model could achieve reconstruction quality comparable to that of classical deep neural network-based solutions. It is worth noting that modern autoencoders based on convolutional neural networks successfully address complex image processing tasks, as discussed in Section 2. Given their capabilities, reconstructing characters from the PolLettDS dataset does not pose a significant challenge for classical models. Consequently, two relatively simple classical autoencoders were selected as benchmarks for evaluating the quantum model. In both cases, it was assumed that the classical autoencoder compresses input data of size 64×64×1 pixels into a latent representation of size 1024, analogous to the quantum model.

The first classical autoencoder (Classical Autoencoder I) had the architecture shown in Table 3. The latent representation was of size 16×16×4, which preserved some spatial structure of the input image. This configuration enabled effective reconstruction using a relatively simple architecture with only 153 parameters.

In the second variant (Classical Autoencoder II), a more aggressive reduction of spatial dimensions of feature maps was applied, resulting in a latent representation in the form of a 1×1×1024 tensor. This led to the complete loss of spatial structure in the input image. This autoencoder consisted of 12 layers and had a total of 1,651,765 parameters. The large number of parameters resulted from the need to use more convolutional and pooling layers to gradually reduce the spatial dimension of feature maps, along with a correspondingly complex decoder to reconstruct the data into the 64×64×1 format. Details of the architecture are provided in Table 4.

Both classical models were trained using the ADAM

optimizer with the mean squared error (MSE) as the loss function. The training was conducted with a learning rate of 0.001, as well as a batch size of 256 and 500 epochs. Additionally, an early stopping mechanism was employed to prevent overfitting.

Examples of the results of the reconstruction of five selected characters, obtained using both classical and two variants of quantum autoencoders, are presented in Fig. 5. This illustration shows the differences in the quality of the reproduction depending on the model type and the size of the hidden space. In the quantum case, the size of the hidden space is the same, but different pairs of qubits for the trash space were chosen. In the case of Quantum Autoencoder I there are qubits (0,1), while in Quantum Autoencoder II the trash space is based on qubits (0,6). As can be seen, this is of significant importance, which can also be noticed on the basis of the average values of the measures presented in Table 5.

Remark 2. Classical images encoded to the form of a quantum register mean that a suitable quantum state which represents a classical image can be entangled. Therefore, if qubits 0 and 1 carry important information about the entanglement and the image, then their choice for the latent/trash space causes significant degradation of the image during the measuring of the latent space. However, as can be seen in selected examples, e.g., digit 4 or 9 in the visual comparison (Fig. 5), the quantum autoencoder is still capable of reconstructing an image with a fairly faithful image representation for the original one.

To quantitatively evaluate reconstruction quality, three metrics were calculated for each test image: the mean squared error (MSE), the structural similarity index (SSI), and quantum fidelity. These results also confirm that classical autoencoders achieve slightly better reconstruction quality, which is consistent with earlier expectations and the current state of quantum computing technology. However, it should be noted that the quantum autoencoders were trained on a much smaller dataset due to the high complexity of parameter optimization. Given this, the quality differences can be considered acceptable.

To deepen the analysis, a second experiment was conducted, in which more test images were reconstructed. For each of the five selected letters (“ć”, “Ę”, “k”, “L”, “Ź”) and two digits (“3”, “7”), a total of seven sets of characters were processed, with 52 samples per character. Average values of all three quality metrics were calculated for each result set. The results are summarized in Table 5.

The analysis of the table confirms the findings from the first experiment. Both classical autoencoders, the simpler (Classical Autoencoder I) and the more complex (Classical Autoencoder II) one, achieved higher fidelity and SSI values and a lower MSE. Among the quantum models, variant II (with the latent space omitting qubits 0 and 6) obteined noticeably better results than variant I,

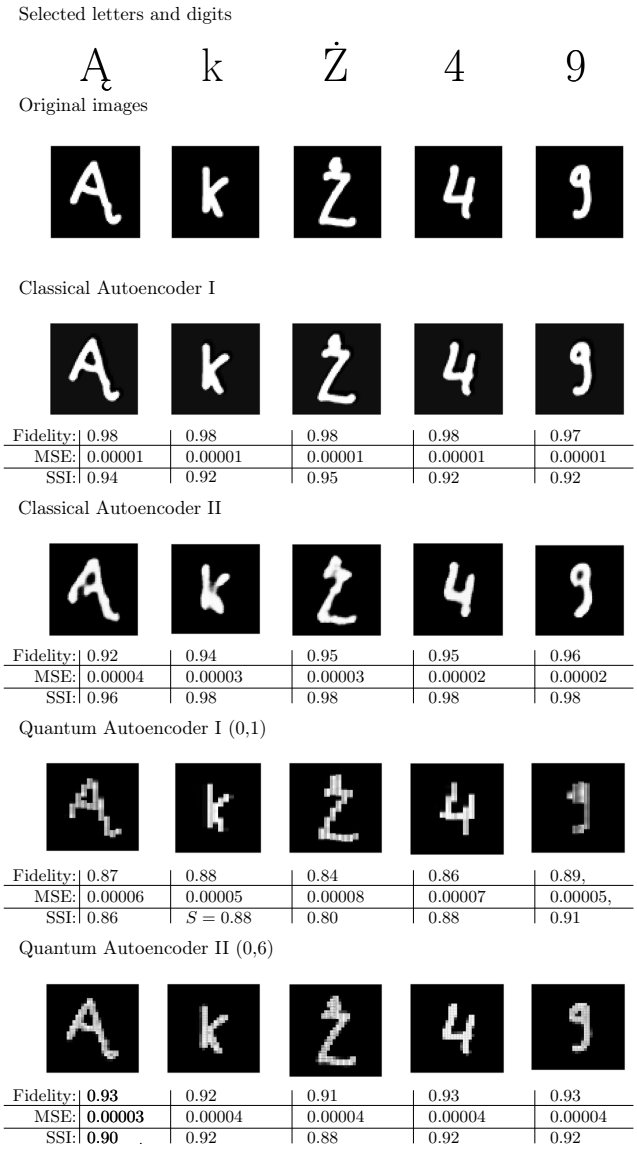

Fig. 5. Visual comparison of the performance quality of selected classical and quantum autoencoders. Five sample characters were selected. The architecture of Classical Autoencoders I and II is presented as Tables 3 and 4, respectively. Quantum autoencoders are based on quantum convolutional networks. In the case of Quantum I (0,1), the latent space was based on qubits 0, 1 while in the second case—on qubits 0 and 6. In quantum autoencoders three layers were used, and a total of 156 parameters describe the encoder and the decoder. The quality measures are fidelity, the MSE (mean squared error), and the SSI (structural similarity index).

which further highlights the importance of qubit selection in the latent space. These differences are particularly evident in the fidelity values, which measure the similarity of quantum states. For Quantum Autoencoder II, the fidelity values approached those of classical autoencoders.

Table 5. Average values of mesures: fidelity (Avg.F.), the mean squared error (Avg.M.) and the structural similarity index (Avg.S.) for two classical autoencoders and a quantum autoencoder (Quantum A.I. (0,1), where the latent space was based on qubits 0 and 1, and for the second case—on 0 and 6 (Quantum A.I. (0,6)).

Sign	Avg.F.	Avg.M.	Avg.S.
Classical A.I			
ć	0.97528	0.00001	0.91748
Ę	0.98223	0.00001	0.94846
k	0.97807	0.00001	0.92891
L	0.97306	0.00001	0.90665
3	0.97996	0.00001	0.93594
7	0.97887	0.00001	0.93354
Ź	0.98181	0.00001	0.94630
Classical A.II			
ć	0.96473	0.00002	0.98612
Ę	0.94045	0.00003	0.97188
k	0.96433	0.00002	0.98518
L	0.97537	0.00001	0.99071
3	0.95626	0.00002	0.98089
7	0.94914	0.00002	0.97743
Ź	0.94762	0.00003	0.97636
Quantum A.I (0,1)			
ć	0.89630	0.00005	0.88250
Ę	0.88674	0.00006	0.84680
k	0.84673	0.00007	0.85406
L	0.85987	0.00007	0.89388
3	0.88603	0.00006	0.87096
7	0.90200	0.00005	0.88187
Ź	0.88477	0.00006	0.85323
Quantum A.II (0,6)			
ć	0.91650	0.00004	0.91261
Ę	0.90844	0.00004	0.87406
k	0.92679	0.00004	0.91021
L	0.91809	0.00004	0.92326
3	0.92113	0.00004	0.90072
7	0.92178	0.00004	0.90216
Ź	0.92016	0.00004	0.88495

Remark 3. It should also be emphasized that Theorem 2 limits the possible quality of reconstruction, which is reflected as well in the value of measures assessing the quality of reconstruction (Fig. 5 and Table 5). However, the reconstruction of the classical image is created by performing a series of measurements on a quantum register obtained from the decoder. Due to the probabilistic nature of this operation, a slight improvement in reconstruction can be achieved in some individual cases. However, by averaging results, and considering Theorem 2, the very design of the quantum autoencoder naturally introduces an upper bound on the quality of reconstruction.

In summary, the experiments demonstrate that the proposed quantum convolutional autoencoder can successfully reconstruct images of letters and digits, achieving reconstruction quality comparable to that of classical solutions while using a significantly smaller training set. The results suggest that this approach may serve as a valuable alternative to classical methods in the future, especially as quantum hardware and learning algorithms continue to improve.

6. Conclusions

This study presented the architecture of a quantum autoencoder based on a convolutional quantum network, designed for efficient reconstruction of handwritten letters and digits. The developed model is grounded in the concept of a quantum convolutional network, enabling the compression of input data and its reconstruction in a form close to the original. The autoencoder was designed to be run on real quantum computers. This was achieved by using gates applied to adjacent qubits, which is an important design consideration for current noisy intermediate-scale quantum (NISQ) hardware, as it facilitates physical implementation of the system. This represents a significant step toward the practical application of quantum machine learning on currently available quantum technologies.

The project takes into account the limitations of contemporary quantum systems, such as the number of available qubits or circuit depth, and the necessity of performing measurement operations. In the conducted experiments, the proposed quantum model was compared with two classical convolutional autoencoders of varying complexity. The reconstruction results, both visual and numerical (MSE, SSI, fidelity), demonstrate that the quantum autoencoder obtains results comparable to those of simpler classical models. This was accomplished despite the limited number of parameters and the use of a significantly smaller training dataset. The experiments also demonstrate that selecting specific qubits to represent the latent space has a significant impact on the quality of reconstructed images. This highlights the importance of carefully designing the quantum circuit structure.

Moreover, a new dataset, named PolLettDS, was developed as part of this research. It contains handwritten digits as well as uppercase and lowercase letters of the Polish alphabet, including diacritical characters. This dataset represents an additional contribution to the development of image processing and handwriting recognition methods for the Polish language. It can also serve as a benchmark for comparing classical and quantum machine learning methods.

The results of this study demonstrate that, despite current technological limitations in quantum

computing, the proposed approach enables effective image reconstruction. In the future, the quantum autoencoder may become a valuable alternative to classical methods. The presented solution, combining theoretical foundations with practical implementation, is another step toward real-world applications of quantum algorithms in image data analysis.

Acknowledgment

We would like to express our gratitude for useful discussions with the *Q-INFO* group at the Institute of Control and Computation Engineering of the University of Zielona Góra, Poland. We also wish to thank the anonymous referees for their useful comments on the preliminary version of the paper. The numerical results were derved using the hardware and software available at GPU μ-Lab located at the Institute of Control and Computation Engineering of the University of Zielona Góra.

References

Al-Khafaji, M. and Ramaha, N. (2025). Hybrid deep learning architecture for scalable and high-quality image compression, *Scientific Reports* **15**(1): 22926, DOI: 10.1038/s41598-025-06481-0.

Aslam, M.M., Tufail, A., Silva, L.C.D., Apong, R.A.A.H.M. and Namoun, A. (2024). An improved autoencoder-based approach for anomaly detection in industrial control systems, *Systems Science & Control Engineering* **12**(1): 2334303, DOI: 10.1080/21642583.2024.2334303.

Baldi, P. and Hornik, K. (1989). Neural networks and principal component analysis: Learning from examples without local minima, *Neural Networks* **2**(1): 53–58, DOI: 10.1016/0893-6080(89)90014-2.

Bravo-Prieto, C. (2021). Quantum autoencoders with enhanced data encoding, *Machine Learning: Science and Technology* **2**(3): 035028, DOI: 10.1088/2632-2153/ac0616.

Brooks, M. (2019). Beyond quantum supremacy: The hunt for useful quantum computers, *Nature* **574**(7776): 19–21, DOI: 10.1038/d41586-019-02936-3.

Cao, C. and Wang, X. (2021). Noise-assisted quantum autoencoder, *Physical Review Applied* **15**(5): 054012, DOI: 10.1103/PhysRevApplied.15.054012.

Ciliberto, C., Herbster, M., Ialongo, A.D., Pontil, M., Rocchetto, A., Severini, S. and Wossnig, L. (2018). Quantum machine learning: A classical perspective, *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* **474**(2209): 20170551, DOI: 10.1098/rspa.2017.0551.

Cong, I., Choi, S. and Lukin, M.D. (2019). Quantum convolutional neural networks, *Nature Physics* **15**(12): 1273–1278, DOI: 10.1038/s41567-019-0648-8.

Cottrell, G.W., Munro, P. and Zipser, D. (1987). Learning internal representation from gray-scale images: An example of extensional programming, *Proceedings of the 9th Annual Meeting of the Cognitive Science Society, Seattle, USA,* pp. 462–473.

Devlin, J., Chang, M., Lee, K. and Toutanova, K. (2018). BERT: Pre-training of deep bidirectional transformers for language understanding, *CoRR* abs/1810.04805.

Du, Y. and Tao, D. (2025). On exploring the potential of quantum auto-encoder for learning quantum systems, *IEEE Transactions on Neural Networks and Learning Systems* **36**(7): 12454–12468, DOI: 10.1109/TNNLS.2024.3474793.

Fournier, Q. and Aloise, D. (2019). Empirical comparison between autoencoders and traditional dimensionality reduction methods, *Proceedings of the IEEE 2nd International Conference on Artificial Intelligence and Knowledge Engineering (AIKE), Los Alamitos, USA,* pp. 211–214, DOI: 10.1109/AIKE.2019.00044.

He, K., Chen, X., Xie, S., Li, Y., Dollár, P. and Girshick, R. (2022). Masked autoencoders are scalable vision learners, *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, USA,* pp. 15979–15988, DOI: 10.1109/CVPR52688.2022.01553.

Higgins, I., Matthey, L., Pal, A., Burgess, C.P., Glorot, X., Botvinick, M. M., Mohamed, S. and Lerchner, A. (2017). β-VAE: Learning basic visual concepts with a constrained variational framework, *Proceedings of the 5th International Conference on Learning Representations, ICLR 2017, Toulon, France*.

Hinton, G.E. and Salakhutdinov, R. (2006). Reducing the dimensionality of data with neural networks, *Science* **313**(5786): 504–507, DOI: 10.1126/science.1127647.

Hirvensalo, M. (2004). *Quantum Computing*, 2nd Edn, Springer, Berlin/Heidelberg, DOI: 0.1007/978-3-662-09636-9.

Huang, H.-Y., Broughton, M., Mohseni, M., Babbush, R., Boixo, S., Neven, H. and McClean, J.R. (2021). Power of data in quantum machine learning, *Nature Communications* **12**(1): 2631, DOI: 10.1038/s41467-021-22539-9.

Janjua, J. and Patankar, A. (2024). Exploring the impact of denoising autoencoder architectures on image retrieval, *Procedia Computer Science* **235**: 2557–2566, DOI: 10.1016/j.procs.2024.04.241.

Kingma, D.P. and Welling, M. (2014). Auto-encoding variational Bayes, *Proceedings of the 2nd International Conference on Learning Representations, ICLR 2014, Banff, Canada*.

Kramer, M.A. (1991). Nonlinear principal component analysis using autoassociative neural networks, *AIChE Journal* **37**(2): 233–243, DOI: 10.1002/aic.690370209.

Kramer, M.A. (1992). Autoassociative neural networks, *Computers & Chemical Engineering* **16**(4): 313–328, DOI: 10.1016/0098-1354(92)80051-A.

Kurowski, K., Slysz, M., Subocz, M. and Różycki, R. (2021). Applying a quantum, annealing based restricted Boltzmann machine for MNIST handwritten

digit classification, *Computational Methods in Science and Technology* **27**(3): 99–107, DOI: 10.12921/cmst.2021.0000011.

Le, Q.V., Ranzato, M., Monga, R., Devin, M., Chen, K., Corrado, G.S., Dean, J. and Ng, A.Y. (2012). Building high-level features using large scale unsupervised learning, *Proceedings of the 29th International Conference on Machine Learning, ICML'12, Madison, USA*, p. 507–514.

Liu, Z., Kalogeiton, V. and Cani, M. (2021). Multiple style transfer via variational autoencoder, *Proceedings of the IEEE International Conference on Image Processing, ICIP 2021, Anchorage, USA,* pp. 2413–2417, DOI: 10.1109/ICIP42928.2021.9506379.

Loey, M., El-Sawy, A. and El-Bakry, H.M. (2017). Deep learning autoencoder approach for handwritten Arabic digits recognition, *CoRR* abs/1706.06720, DOI: 10.48550/arXiv.1706.06720.

Luhman, T. and Luhman, E. (2023). High fidelity image synthesis with deep VAEs in latent space, *CoRR* abs/2303.13714, DOI: 10.48550/ARXIV.2303.13714.

Ma, H., Huang, C.-J., Chen, C., Dong, D., Wang, Y., Wu, R.-B. and Xiang, G.-Y. (2023). On compression rate of quantum autoencoders: Control design, numerical and experimental realization, *Automatica* **147**: 110659, DOI: 10.1016/j.automatica.2022.110659.

Makhzani, A., Shlens, J., Jaitly, N. and Goodfellow, I.J. (2015). Adversarial autoencoders, *CoRR* abs/1511.05644, DOI: 10.48550/arXiv.1511.05644.

Mangini, S., Marruzzo, A., Piantanida, M., Gerace, D., Bajoni, D. and Macchiavello, C. (2022). Quantum neural network autoencoder and classifier applied to an industrial case study, *Quantum Machine Intelligence* **4**(2): 13, DOI: 10.1007/s42484-022-00070-4.

Masci, J., Meier, U., Cireşan, D. and Schmidhuber, J. (2011). Stacked convolutional auto-encoders for hierarchical feature extraction, *in* T. Honkela *et al.* (Eds), *Artificial Neural Networks and Machine Learning, ICANN 2011*, Springer, Berlin/Heidelberg, pp. 52–59, DOI: 10.1007/978-3-642-21735-7_7.

Neloy, A.A. and Turgeon, M. (2024). A comprehensive study of auto-encoders for anomaly detection: Efficiency and trade-offs, *Machine Learning with Applications* **17**: 100572, DOI: 10.1016/j.mlwa.2024.100572.

Nielsen, M.A. and Chuang, I.L. (2010). *Quantum Computation and Quantum Information: 10th Anniversary Edition*, Cambridge University Press, Cambridge.

Parthasarathy, K. (1992). *An Introduction to Quantum Stochastic Calculus*, Birkhäuser, Basel, DOI: 10.1007/978-3-0348-0566-7.

Powell, M.J.D. (1994). A direct search optimization method that models the objective and constraint functions by linear interpolation, *in* S. Gomez and J.P. Hennart (Eds), *Advances in Optimization and Numerical Analysis,* Mathematics and Its Applications, Vol. 275, Springer Netherlands, Dordrecht, pp. 51–67, DOI: 10.1007/978-94-015-8330-5_4.

Powell, M.J.D. (2007). A view of algorithms for optimization without derivatives, *Technical report*, DAMTP 2007/NAO3, Cambridge University, Cambridge.

Preskill, J. (2018). Quantum computing in the NISQ era and beyond, *Quantum* **2**: 79, DOI: 10.22331/q-2018-08-06-79.

Ritter, M.B. (2019). Near-term quantum algorithms for quantum many-body systems, *Journal of Physics: Conference Series* **1290**(1): 012003, DOI: 10.1088/1742-6596/1290/1/012003.

Romero, J., Olson, J.P. and Aspuru-Guzik, A. (2017). Quantum autoencoders for efficient compression of quantum data, *Quantum Science and Technology* **2**(4): 045001, DOI: 10.1088/2058-9565/aa8072.

Sawerwain, M. and Kowal, M. (2025a). Github repository for the PolLettDS dataset, `https://github.com/qMSUZ/PolLettDS`.

Sawerwain, M. and Kowal, M. (2025b). Source code of examples of a quantum autoencoder, `https://github.com/qMSUZ/QAutoEnc`.

Shang, W., Qiu, J., Shi, H., Wang, S., Ding, L. and Xiao, Y. (2024). An efficient anomaly detection method for industrial control systems: Deep convolutional autoencoding transformer network, *International Journal of Intelligent Systems* **2024**(1): 5459452, DOI: 10.1155/2024/5459452.

Sivarajah, S., Dilkes, S., Cowtan, A., Simmons, W., Edgington, A. and Duncan, R. (2020). tlket⟩: A retargetable compiler for NISQ devices, *Quantum Science and Technology* **6**(1): 014003, DOI: 10.1088/2058-9565/ab8e92.

Slysz, M., Kurowski, K., Waligóra, G. and Węglarz, J. (2023). Exploring the capabilities of quantum support vector machines for image classification on the mnist benchmark, *in* J. Mikyška *et al.* (Eds), *Computational Science, ICCS 2023*, Springer Nature Switzerland, Cham, pp. 193–200, DOI: 10.1007/978-3-031-36030-5_15.

Tokovarov, M., Kaczorowska, M. and Milosz, M. (2020). Development of extensive Polish handwritten characters database for text recognition research, *Advances in Science and Technology Research Journal* **14**(3): 30–38, DOI: 10.12913/22998624/122567.

Vincent, P., Larochelle, H., Bengio, Y. and Manzagol, P.-A. (2008). Extracting and composing robust features with denoising autoencoders, *Proceedings of the 25th International Conference on Machine Learning, ICML'08, New York, USA,* pp. 1096–1103, DOI: 10.1145/1390156.1390294.

Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y. and Manzagol, P.-A. (2010). Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion, *Journal of Machine Learning Research* **11**: 3371–3408.

Wang, H., Tan, J., Huang, Y. and Zheng, W. (2024). Quantum image compression with autoencoders based on parameterized quantum circuits, *Quantum Information Processing* **23**(2): 41, DOI: 10.1007/s11128-023-04243-3.

Weigold, M., Barzen, J., Leymann, F. and Salm, M. (2021). Expanding data encoding patterns for quantum algorithms, *Proceedings of the IEEE 18th International Conference on Software Architecture Companion (ICSA-C), Stuttgart, Germany,* pp. 95–101, DOI: 10.1109/ICSA-C52384.2021.00025.

Wu, J., Fu, H., Zhu, M., Zhang, H., Xie, W. and Li, X.-Y. (2024). Quantum circuit autoencoder, *Physical Review A* **109**(3): 032623, DOI: 10.1103/PhysRevA.109.032623.

Zeguendry, A., Jarir, Z. and Quafafou, M. (2023). Quantum machine learning: A review and case studies, *Entropy* **25**(2): 287, DOI: 10.3390/e25020287.

Marek Sawerwain received his ME degree in 2004 and his PhD degree in 2010 from the Faculty of Electrical Engineering, Computer Science and Telecommunications of University of Zielona Góra, Poland. He obtained his DSc (habilitation) in computer science from the Technical University of Silesia in 2021. His current scientific interests include quantum communication and methods, and the theory of quantum programming languages. He also conducts his research in the field of quantum computations models simulations and the development of effective algorithms for solutions based on modern multicore CPU and GPU technology. He is currently with the Faculty of Engineering and Technical Sciences, University of Zielona Góra, as an associate professor.

Marek Kowal received his PhD in electrical engineering from the University of Zielona Góra, Poland, in 2004, and his DSc (habilitation) in computer science from the Częstochowa University of Technology in 2020. He is currently an associate professor at the Institute of Control and Computation Engineering of the University of Zielona Góra. His research interests focus on deep neural networks and self-supervised learning, particularly applied to multiple object tracking and medical image analysis.

Józef Korbicz has been a full-rank professor of automatic control at the University of Zielona Góra, Poland, since 1994. In 2007 he was elected a corresponding member and in 2020 a full member of the Polish Academy of Sciences. His current research interests include fault detection and isolation, control theory and computational intelligence. He has published over 490 scientific works, authored or co-authored eight books and coedited 28 books. He served as the IPC chairman of the IFAC Safeprocess Symposium held in Beijing, China, in 2006, and as the chairman of the NOC for *Safeprocess* held in Warsaw, Poland, in 2018. He is currently the chair of the Scientific Council of the Systems Research Institute of the Polish Academy of Sciences, a senior member of the IEEE, a member of the IFAC *Safeprocess* TC and the chair of the Commission for Computer Science and Automation of the Polish Academy of Sciences, Poznań Branch. More: `http://www.uz.zgora.pl/~jkorbicz`.

Appendix

Unitary evolution and Stone's theorem

For readers' convenience, we recall some basic information on unitray evolution and Stone'a theorem, which are used in our Theorem 1 on quantum autoencoder existence.

A1. Unitary evolution

The postulates of quantum mechanics, and quantum computation as well, about the existence of a unitary operator which transforms input quantum state into output state, used by us in the proof of the existence of a quantum autoencoder, refer to the notion of the following function U_t dependent on time t:

$$U_t : \mathcal{H}_n \to \mathcal{H}_n. \tag{A1}$$

Therefore, if we regarded a new quantum state of the system to be denoted by q_t, then, for the initial state q_0,

$$q_t = U_t q_0. \tag{A2}$$

Then, a map U_t (or the function U_t) is regarded as unitary evolution with the following conditions:

(i) For $t \in \mathbb{R}$ and $\psi \in \mathcal{H}_n$, $||U_t\psi|| = ||\psi||$.
This condition means than the norm of quantum state is preserved during unitary evolution.

(ii) The map U_t is linear,

$$\begin{aligned}\forall_t \;\; & U_t(\alpha_1|\psi_1\rangle + \alpha_2|\psi_2\rangle + \ldots + \alpha_n|\psi_n\rangle) \\ & = \alpha_1 U_t|\psi_1\rangle + \alpha_2 U_t|\psi_2\rangle + \ldots + \alpha_n U_t|\psi_n\rangle,\end{aligned} \tag{A3}$$

i.e., operation U_t acts on the base states of the system $|\psi\rangle$ independently.

(iii) For $t_1, t_2 \in \mathbb{R}$, $U_{t_1+t_2} = U_{t_1} \cdot U_{t_2}$.

(iv) The condition (iii) requires continuous time evolution:

$$\forall t_0 \in \mathbb{R}, \lim_{t\to t_0} U_t\psi(0) = \lim_{t\to t_0} \psi(t) = \psi(t_0). \tag{A4}$$

The conditions (i)–(iii) are usually summed up with the following lemma.

Lemma A1. (Hirsenvalo, 2004) *A map U_t describing the time evolution of a system satisfying the conditions (i), (ii) and (iii) is represented by a group of unitary operators.*

A2. Stone's theorem

Taking into account the condition (iv) it is possible to form the following theorem (a detailed discussion of this theorem can be found, e.g., in the work of Parthasarathy (1992)).

Theorem A1. (Stone's theorem) *For every map U_t which meets the conditions (i), (ii), (iii) and (iv), there exists only one self-adjoint operator H fulfilling the relation*

$$U_t = e^{-itH}. \tag{A5}$$

Stone's theorem allows us to write time evolution using the following equation:

$$\psi(t) = e^{-itH}\psi(0). \tag{A6}$$

After two-side differentiation, the so-called the Schrödinger equation is obtained:

$$i\frac{d}{dt}\psi(t) = H\psi(t). \tag{A7}$$

Remark A1. The evolution discussed briefly in this appendix is continuous. However, in the case of autoencoder systems studied in this paper, evolution is presented only in discrete points of time. Therefore, the state of a system will be given in discrete points of time $t_0, t_1, t_2, \ldots, t_n$, after, e.g., application of successive QC, QP operators, and the value t_n denotes the point of time where the computational process is completed.

Due to that, the change of a state will be described as a sequence of vectors with suitable unitary operators: $U_1\psi$, $U_2U_1\psi$, $U_3U_2U_1\psi$, etc. Stone's theorem, (especially Eqns. A6 and A7 there) shows the fact that between two states $|\psi_i\rangle$, $|\psi_{i+1}\rangle$ always exists a unitary operator which transforms $|\psi_i\rangle$ into $|\psi_{i+1}\rangle$.

Received: 21 July 2025
Revised: 1 October 2025
Accepted: 27 October 2025