

Comparing Emerging Technologies in Image Classification: From Quantum to Kolmogorov

Fabio Napoli(✉), Mariarosaria Castaldo(✉), Stefano Marrone(✉), and Lelio Campanile(✉)

Dipartimento di Matematica e Fisica, Università della Campania "Luigi Vanvitelli", Caserta 81100, Italy
{fabio.napoli,mariarosaria.castaldo,stefano.marrone, lelio.campanile}@unicampania.it

Abstract. The rapid evolution of Artificial Intelligence has led to significant advancements in image classification, with novel approaches emerging beyond traditional deep learning paradigms. This paper presents a comparative analysis of three distinct methodologies for image classification: classical Convolutional Neural Networks (CNNs), Kolmogorov-Arnold Networks (KANs) and KAN-based CNNs and Quantum Machine Learning using Quantum Convolutional Neural Networks. The study evaluates these models on the Labeled Faces in the Wild dataset, implementing the different classifiers with existing, well-assessed technologies. Given the fundamental differences in computational paradigms, performance assessment extends beyond traditional accuracy metrics to include computational efficiency, interpretability, and, for quantum models, gate depth and noise.

As a summary of the results, the proposed Quantum Convolutional Neural Network (QCNN) model achieves an accuracy of 75% on the target images classification task, indicating promising performance within current quantum computational limits. All the experiments strongly suggest that Convolutional Kolmogorov-Arnold Networks (CKANs) exhibit increased accuracy as image resolution decreases, QCNN performance meaningfully changes in relation to noise level, while CNNs still keeping strong discriminative capabilities.

Keywords: Kolmogorov-Arnold Networks · Quantum Convolutional Neural Networks · Image Classification

1 Introduction

Image classification is a cornerstone of Computer Vision (CV) and Machine Learning (ML), with applications spanning numerous fields, from medical diagnosis to autonomous driving. Classical CNNs have consistently achieved state-of-the-art performance in image classification tasks [8,18]. Their ability to learn hierarchical features from raw pixel data has revolutionized image recognition.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2026
O. Gervasi et al. (Eds.): ICCSA 2025 Workshops, LNCS 15886, pp. 260–273, 2026.
https://doi.org/10.1007/978-3-031-97576-9_17

However, the computational demands of training deep CNNs, especially for complex datasets, remain a significant challenge. Furthermore, classical CNNs can be vulnerable to adversarial attacks, raising concerns about their robustness and reliability [8].

Hence, CNNs' limitations have spurred research into alternative architectures. Quantum Machine Learning (QML) offers a potential avenue for enhancing or even surpassing classical methods. QCNNs leverage quantum phenomena, such as superposition and entanglement, to potentially speed up computation and improve feature extraction. Another class of networks—the KANs—offers an alternative approach to Deep Learning (DL). They construct a hierarchical representation of the data by composing simple functions.

Comparing these different architectures—classical CNNs, QCNNs, and KANs—presents several key challenges.

- Fair Comparison: Establishing a fair comparison framework across diverse architectures requires careful consideration of factors such as dataset complexity, computational resources, and evaluation metrics.
- QCNNs are increasingly studied as candidates for image classification tasks, yet achieving both scalability and efficiency in their design remains nontrivial. The introduced limitations are well analysed by [18]. Moreover, the constraints imposed by present-day quantum hardware significantly affect how these architectures can be structured and realized in practice.
- KAN Applicability to Image Data: while KANs have shown promise in other domains, their effectiveness for image classification requires further investigation. Adapting KANs to handle the high dimensionality and spatial structure of image data presents a unique challenge.
- Interpretability and Explainability: Understanding the inner workings of both QCNNs and KANs, especially in the context of image classification, is crucial for gaining insights into their decision-making processes and building trust in their predictions.

This paper aims to provide a comprehensive comparison of classical CNNs, QCNNs, and KANs for image classification. We will evaluate their performance on benchmark datasets, analyse their computational complexity, and investigate their robustness to noise and adversarial perturbations. By systematically exploring the strengths and weaknesses of each architecture, this research contributes to a better understanding of their capabilities and limitations for image recognition tasks. More concretely, this study evaluates three models—CNNs, QCNNs, and CKANs—on the Labelled Faces in the Wild (LFW) dataset, implementing the different classifiers with existing well-assessed technologies and software libraries. The models are compared against classical performance indices, i.e., accuracy and loss.

The paper is structured as follows: Sect. 2 reports related scientific papers; Sect. 3 describes the architecture of the chosen models and some technical details; Sect. 3.5 reports the results of the experiments that are discussed in Sect. 4; Sect. 5 ends the paper drawing conclusions and future work.

2 Related Works

Image classification has significantly evolved, with classical CNNs establishing themselves as the benchmark approach over the last decade. CNNs use hierarchical structures to efficiently extract spatial features and achieve superior performance across various benchmarks, especially in face recognition tasks such as the LFW dataset [7]. Landmark CNN architectures such as AlexNet [11], VGGNet [17], and ResNet [6] have progressively enhanced classification accuracy and generalization. Recent studies [5,14] remark on CNNs strengths, including their capability to model complex visual patterns, alongside ongoing challenges such as computational cost and vulnerability to adversarial attacks [4].

KANs, recently proposed in [13], represent an innovative alternative to conventional neural networks by employing spline-based univariate functions that significantly improve interpretability and computational efficiency. Although primarily validated in numerical and scientific contexts [13], recent works have begun applying KAN structures within convolutional frameworks CKANs, demonstrating promising results in remote sensing [1] and general CV tasks. These studies suggest CKANs could achieve competitive accuracy compared to classical CNNs, with the additional benefits of improved interpretability and reduced model complexity.

QML, and in particular QCNNs, provide a distinct computational paradigm exploiting quantum mechanics for enhanced data processing [16]. QCNNs, first outlined in [2], exploit quantum circuits for convolution-like operations, demonstrating promising classification capabilities even on current noisy quantum hardware [9,18]. Studies comparing QCNNs with classical CNNs suggest potential quantum advantages in efficiency and parameter reduction, although noise robustness remains a significant limitation [8].

Direct comparative studies of these advanced ML methodologies—CNN, KAN, and QCNN—remain scarce. However, isolated comparisons suggest each approach offers unique advantages: CNNs excel in absolute accuracy and maturity, KANs provide enhanced interpretability and efficiency, and QCNNs offer promising future potential subject to hardware advancements [1,18]. The present study aims to bridge this gap systematically by evaluating these methodologies under unified experimental conditions, thus contributing to a comprehensive understanding of their relative strengths and applicable domains.

3 Methodology and Experimentation

3.1 Dataset Description

The LFW dataset is a fundamental resource for the study of unconstrained face recognition [7]. Created in 2007, it contains over 13,000 images collected from the web, each labelled with the name of the depicted subject.

This dataset provides greyscale images at a low resolution (62×47 pixels), making it easy to explore and experiment with ML algorithms.

The dataset has supported the development of numerous face recognition techniques, including models based on convolutional networks, Support Vector Machine (SVM), and Principal Component Analysis (PCA). Data can be tailored to specific computational needs, enabling time-quality tradeoffs.

The dataset's inherent diversity in terms of illumination, facial expressions, and poses has made it a useful tool for comparing different methods and is an ideal dataset for academic research on facial verification. However, it has some weaknesses: the demographic representation is unbalanced, and it does not support extreme photographic conditions. These shortcomings make it less suitable for operational scenarios that require robustness to low-light conditions or significant occlusions [7].

LFW has two main advantages: the first is the integration with Scikit-learn, which simplifies the entire experimental workflow, allowing developers to directly test their models; the second is the presence of metadata associated with the images that helps analyze the performance of different approaches. With these advantages, LFW represents a solid starting point for facial recognition research in the broader Artificial Intelligence (AI) field.

3.2 The CNN Model

CNNs are a class of feedforward neural networks commonly used to process and make predictions on different types of data such as text, images, video, and audio [12].

Thanks to their architecture that makes them particularly effective in image classification and object detection [12], they can capture spatial relationships from the data as they are characterized by convolution filters that analyze the characteristics of the input data to provide answers.

They are formed by an input layer, an output layer, and hidden layers, some of which perform convolutions. The layers are distinguished as:

- the *convolutional layer* applies filters to identify local features in the data, such as edges and textures [3];
- the *pooling layer* reduces the spatial dimensions of the input images through operations such as max pooling, which improves the robustness and generalization capabilities of the network [3];
- the *fully connected layer* processes the features extracted from the previous layers, obtaining a final classification using an activation function: the softmax function is often used [3].

The construction of this type of network involves an initial phase of data pre-processing; after defining the parameters for the network structure and by using non-linear activation functions, such as Rectified Linear Unit (ReLU), the training can learn the complex relationships between data.

Subsequently, the optimization phase is planned using algorithms such as Adam and regularization techniques such as dropout and batch normalization to prevent the overfitting phenomenon [10,15];

3.3 KAN and CKAN Network

KANs are a promising alternative to Multilayer Perceptrons (MLPs), developed to efficiently and interpretably approximate multivariate functions. They are based on the Kolmogorov-Arnold theorem, which shows that a continuous multivariate function on a bounded domain can be written as a finite composition of continuous functions of a single variable, and on the binary addition operation [13]. Thanks to this theorem, the KANs architecture is particularly suitable for applications that require high precision.

These neural networks have a fully connected structure like MLPs, but unlike the latter, KANs apply activation functions on the edges of the network; therefore, they do not use linear weight matrices, since each weight parameter is replaced by a spline function [13]. Each function is defined by spline coefficients and a grid of control points that dynamically adapts to the data during training, thus improving the accuracy of the [13] model. In addition, splines can also be used with activation functions—e.g., Sigmoid Linear Unit (SiLU)—in the case of complex relationships. The nodes of KAN networks are classified into:

- *additive nodes* that add the outputs of the spline functions of the previous layers [13];
- *multiplicative nodes* that multiply the contributions to capture nonlinear relationships between the data [13].

These types of nodes can be combined in different ways to improve the capacity of the network. [13].

An important aspect of KANs is their ability to integrate symbolic components that allow them to explicitly represent already known mathematical functions, such as logarithms or powers [13]. In this way, KANs offer a level of interpretability that other deep learning models struggle to achieve, making them particularly suitable for contexts where transparency and model understanding are key requirements [13].

The learning process in KANs uses automatic optimization by differentiation, an approach that allows flexible combinations of different objectives: starting from a Mean Squared Error (MSE), L1 regularization elements and entropy-based penalties are added, with the aim of pushing the network to generate simpler and more readable models [13].

KANs can be trained with common optimizers like Adam, but it also supports more sophisticated methods, like Limited-memory BFGS (LBFGS), which are particularly effective when working with limited datasets [13].

Another interesting aspect is the possibility of simplifying the network by removing unimportant nodes that do not contribute, through pruning techniques, to reduce the computational load without compromising performance [13].

These networks are also characterized by an automatic saving system that allows to keep track of any changes made to the model, storing the state of the network after the application, for example, of the pruning technique or the redefinition of the spline grid. This mechanism simplifies the management of experiments and protects against possible data loss [13].

KANs are particularly suitable for modeling physical phenomena, forecasting time series, or scientific applications where the quality of interpretation is as important as numerical accuracy, representing an advanced and versatile solution in the landscape of artificial intelligence and machine learning [13].

The Pykan library[1] is a framework designed for implementing and training KANs [13]. Experiments were conducted using CKAN networks, an architecture that incorporates KANs, as they have demonstrated superior performance in image classification tasks.

The CKAN architecture represents a fusion of the CNNs and KANs networks: convolutional layers extract spatial features from input images, while KAN layers replace traditional fully connected layers with spline functions [1].

This hybrid architecture achieves an optimal balance between computational efficiency and representation power, making it ideal for advanced computer vision applications [1].

3.4 Quantum Machine Learning

In this part, we describe the approach used to design and evaluate the QCNN architecture for image classification. The process consists of several essential steps, including data preprocessing, mapping classical data into quantum states, constructing the quantum convolutional layers, and incorporating realistic noise models to reflect hardware imperfections. These experiments have been conducted using `PennyLane` to simulate quantum circuits, while `Scikit-learn` is used for the preparation and handling of classical data. This methodology aims to explore both the strengths and current limitations of QCNNs in ideal conditions and under the influence of quantum noise, following core principles of quantum machine learning as discussed in prior literature [16]

As described in Sect. 3.1, we adopt the LFW dataset [7] for our experiments. To ensure a balanced and computationally feasible task, we focus on a binary classification problem distinguishing between the identities "Colin Powell" and "George W. Bush", selecting only those classes with at least 200 images. All images are converted to greyscale and normalized to the $[0, 1]$ range before quantum encoding.

To adapt the dataset to the constraints of current quantum hardware, the dimensionality is reduced by using PCA. This transformation projects the data onto a lower-dimensional subspace while preserving the most significant features. Mathematically, this is expressed as in Eq. 1,

$$X_{\text{PCA}} = X \cdot W_{\text{PCA}}, \tag{1}$$

where X denotes the original data matrix and W_{PCA} contains the eigenvectors corresponding to the largest eigenvalues of the covariance matrix of X. For quantum encoding, we use angle embedding to map classical data vectors $x \in \mathbb{R}^d$ (after PCA) into quantum states via rotations around the X-axis, as in Eq. 2,

[1] The library is available on GitHub at https://github.com/KindXiaoming/pykan.

$$|\psi\rangle = \prod_{i=1}^{d} R_X(x_i)|0\rangle, \tag{2}$$

where $R_X(x_i)$ is a single-qubit rotation gate parameterized by the i-th feature. This approach ensures that the input data is efficiently represented in the quantum Hilbert space.

The QCNN architecture is built from repeated layers of parameterized gates and entangling operations. Each layer consists of a sequence of single-qubit rotations and Controlled NOT (CNOT) gates, described by the unitary transformation:

$$U_{\text{layer}}(\theta) = \prod_{i=1}^{d} \text{CNOT}(i, (i+1) \bmod d) \cdot \prod_{i=1}^{d} R_Z(\theta_{i,1}) R_Y(\theta_{i,2}) R_Z(\theta_{i,3}), \tag{3}$$

where $\theta_{i,j}$ are trainable parameters. To improve generalization and prevent overfitting, a dropout is integrated into the quantum circuit by applying depolarizing channels after each layer. The effect of dropout on a quantum state ρ is modelled as in Eq. 4.

$$\mathcal{E}_{\text{dropout}}(\rho) = (1-p)\rho + \frac{p}{2^n} I, \tag{4}$$

where p is the dropout probability, n is the number of qubits, and I is the identity matrix.

To simulate real-world hardware imperfections, we also included depolarizing and phase damping noise channels after each layer. These are given by Eq. 5 and Eq. 6,

$$\mathcal{E}_{\text{depolarizing}}(\rho) = (1-p)\rho + \frac{p}{3}(X\rho X + Y\rho Y + Z\rho Z), \tag{5}$$

$$\mathcal{E}_{\text{phase damping}}(\rho) = (1-p)\rho + pZ\rho Z, \tag{6}$$

which model decoherence and environmental effects in noisy intermediate-scale quantum devices. We adopt a cost function that combines MSE with Norm L2 (L2) regularization as in Eq. 7,

$$\mathcal{L}(\theta) = \frac{1}{N} \sum_{i=1}^{N} (y_i - f(x_i; \theta))^2 + \lambda \|\theta\|_2^2, \tag{7}$$

where y_i is the true label, $f(x_i; \theta)$ is the model prediction, and λ is the regularization coefficient. Optimization is performed using the Adam algorithm with a decaying learning rate, defined at epoch t by 8,

$$\eta_t = \frac{\eta_0}{\sqrt{t} + \epsilon}, \tag{8}$$

where η_0 is the initial learning rate and ϵ is a small constant for numerical stability.

This methodology constitutes a complete pipeline for the development and evaluation of QCNNs in both ideal and noisy environments. The inclusion of noise models and dropout mechanisms provides insights into the feasibility and robustness of deploying quantum models on real hardware.

Figure 1 shows the structure of a single QCNN layer, from angle embedding through trainable gates and noise injection.

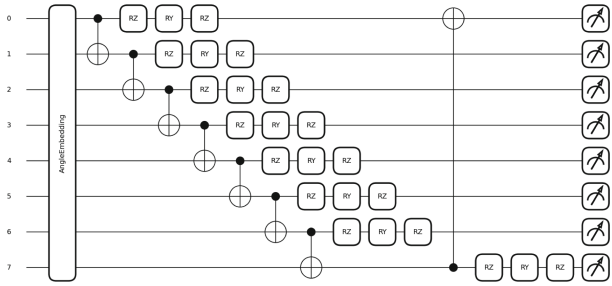

Fig. 1. Quantum circuit representation of a single QCNN layer.

3.5 Experimentation

This section analyses and compares three distinct approaches to image classification, evaluating the performance of a CNN, a hybrid architecture integrating CNN with KAN layers (CKAN), and a QCNN. The objective is to assess the effectiveness of each model in feature extraction and processing, highlighting the advantages and limitations of each methodology.

The CNN implementation was carried out using the LFW dataset, preprocessed with images resized to a scale of 0.4 and normalized based on the dataset's mean and standard deviation.

The images were reshaped into the format (`n_samples,1,h,w`) to explicitly include the channel and preserve spatial structure, ensuring compatibility with CNN requirements.

The model architecture consists of three convolutional blocks with 3×3 kernels, followed by batch normalization, ReLU activation, MaxPooling, and dropout (0.25). The number of filters in the convolutional blocks progressively increases to extract hierarchical image features. The convolutional output is then

flattened and passed through two fully connected layers, with ReLU activation in the first layer and an output dimension corresponding to the number of classes.

Training was conducted using the *CrossEntropyLoss* function and the Adam optimizer (lr=0.001), with a dataset split into 70% training and 30% testing. The process was run for up to 100 epochs, with early stopping based on validation loss. Accuracy and loss metrics were monitored (Fig. 2) to evaluate model performance, ensuring a thorough analysis of the network's behaviour.

The experiment was also conducted on an architecture that combines a CNN with KAN layers (Fig. 3), designed to model advanced nonlinearities using spline-based transformations.

The images were normalized, resized to explicitly include the channel, and split into training (70%) and test (30%) sets. The data was then converted into tensors and loaded using *DataLoader*.

The implemented CKAN model consists of two convolutional layers followed by max pooling operations, progressively reducing spatial dimensionality. The convolutional output, transformed into a linear format, is processed by KAN layers, enabling the model to learn complex data relationships. The output size after pooling is dynamically computed to ensure compatibility with subsequent layers.

This combination of CNN and KAN leverages the CNNs feature extraction capabilities along with the flexibility of spline-based transformations, making the model more interpretable and suitable for analysing structured and complex data.

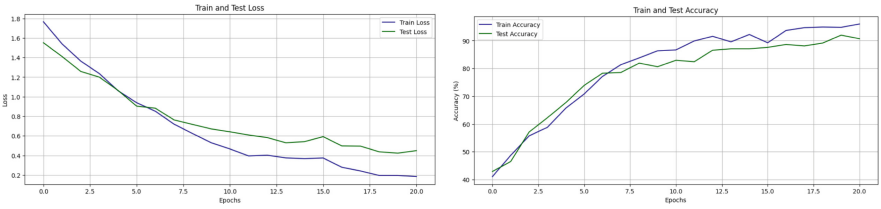

Fig. 2. CNN: training and test loss (left) and accuracy (right).

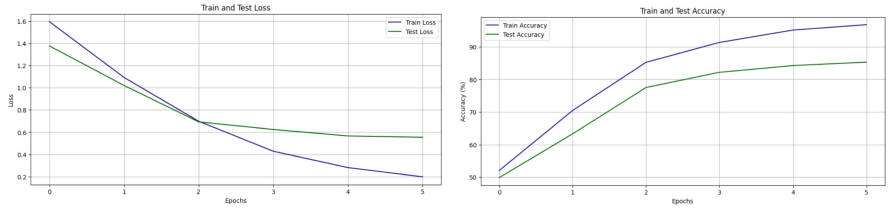

Fig. 3. CKAN: training and test loss (left) and accuracy (right)

Complementing the classical and hybrid approaches, the QCNN was evaluated under both noise-free (Fig. 4) and noisy (Fig. 6) conditions to explore the capabilities and current limitations of quantum-enhanced models.

The LFW dataset was preprocessed using Scikit-learn by normalizing pixel values and flattening images into feature vectors. PCA was applied to retain eight principal components, scaled for quantum embedding. The data was split into training (60%) and testing (40%) sets, maintaining class balance.

QCNNs were implemented in PennyLane using ten layers, each composed of parameterized single-qubit rotations and CNOT entanglement. Angle embedding mapped each 8-dimensional input to quantum states, and Pauli-Z expectation values were used for output measurement.

Training used the Adam optimizer with a learning rate of 0.05. The cost function combined MSE and L2 regularization. Simulations were run for both 5 and 10 epochs using the `default.qubit` (ideal) and `default.mixed` (noisy) backends. The noise model included depolarizing (0.01) and phase damping (0.02) channels.

Each noiseless layer used 40 gates, totaling 240 parameters. In the noisy setting, two additional noise operations per qubit increased the gate count to 72 per layer.

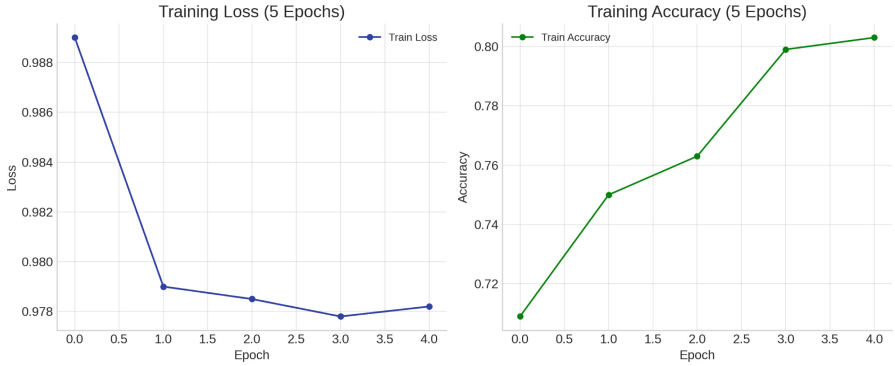

Fig. 4. Noise-free QCNN: training loss (left) and accuracy (right) after 5 epochs.

The full implementation of all models discussed in this work, including the QCNN, CNN, and CKAN architectures, is available on GitHub[2].

4 Discussion

The comparative analysis of classical, hybrid, and quantum neural network models applied to the LFW offers a multifaceted perspective on model behaviour, learning dynamics, and practical trade-offs. As expected, CNNs exhibit strong

[2] https://github.com/fabionpl/ICCSA25.

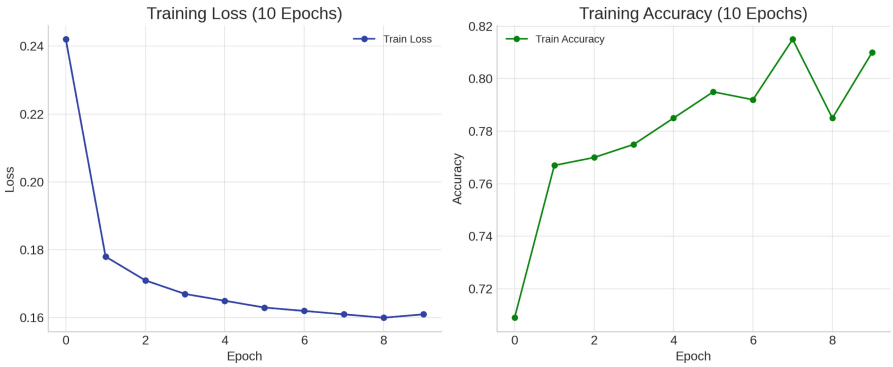

Fig. 5. Noise-free QCNN: training loss (left) and accuracy (right) after 10 epochs.

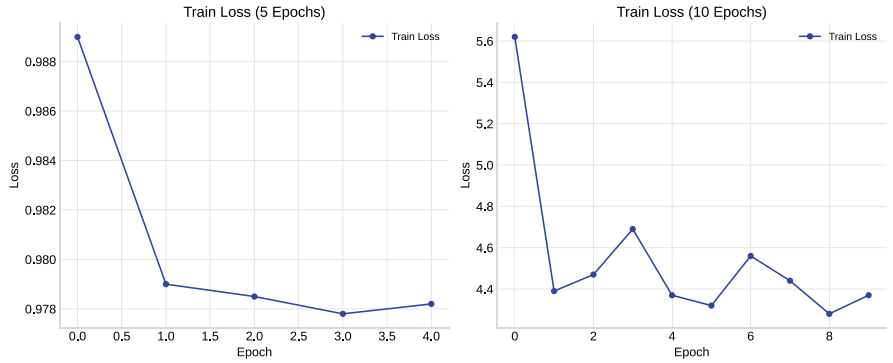

Fig. 6. Noisy QCNN: training loss after 5 epochs (left) and training loss after 10 epochs (right).

discriminative capabilities, consistently reaching high levels of test accuracy. However, this performance often comes at the cost of longer training times and susceptibility to overfitting, especially when training is prolonged or input dimensions are increased. The integration of KAN within a convolutional architecture, as realized in the KAN-CNN model, provides an intriguing alternative. When evaluated across different image resizing settings, KAN-CNN maintains stable test losses while exhibiting improved accuracy as image resolution decreases. This counterintuitive trend suggests that the model benefits from reduced input complexity, perhaps due to a better alignment between its functional decomposition and the compressed visual features. Furthermore, its significantly shorter training times and faster convergence, compared to the classical CNN, reinforce its applicability in contexts where computational efficiency is essential. Although the CNN outperforms KAN-CNN in absolute terms, especially with medium-resolution inputs, the hybrid model remains competitive, particularly when accuracy must be balanced with inference speed and resource constraints.

Turning to the quantum domain, the QCNN introduces a fundamentally different paradigm. In the absence of noise, the QCNN is capable of learning non-trivial representations, achieving a training accuracy of approximately 75%.

Although the quantum model did not surpass the performance of classical networks, its outcomes remain significant considering the nascent state of quantum hardware and its current limitations. However, the presence of quantum noise substantially impacts learning, resulting in minimal improvement in accuracy and erratic loss patterns across epochs. This clear disparity between noiseless simulations and realistic conditions highlights a major hurdle in quantum machine learning: the urgent need for architectures and training methods that can tolerate noise effectively. In general, the comparison reveals a nuanced balance between strengths and weaknesses. Although CNNs continue to lead in raw predictive power, hybrid approaches like KANs-CNNs demonstrate strong potential in scenarios requiring efficient training and adaptability to lower resolution input. Quantum networks, albeit still in development, represent an exciting frontier, with their full utility likely to emerge alongside advancements in hardware. Ultimately, choosing the appropriate model depends on the specific requirements of the task, whether that be accuracy, explainability, computational efficiency, or future scalability.

5 Conclusion and Future Directions

This work offers a comparative analysis of three innovative approaches to image classification: CNNs, CKANs, and Quantum Neural Networks (QNNs), applied to the well-known LFW dataset. CNN-based models confirmed their effectiveness and maturity, consistently achieving high accuracy and proving suitable for real-world deployment. In contrast, KAN-based architectures, particularly CKANs, demonstrated notable advantages in computational efficiency and model interpretability, making them appealing in scenarios where transparency and reduced computational cost are critical.

On the other hand, QCNNs offered intriguing potential, demonstrating their capabilities under ideal, noiseless scenarios, but highlighting notable limitations under realistic, noisy quantum conditions.

These results have significant practical implications. CNNs remain the benchmark for applications requiring maximum predictive performance and proven reliability, especially in real-world environments. CKAN architectures offer efficient training and improved interpretability without significantly compromising accuracy. QCNNs, still constrained by hardware limitations and noise sensitivity, offer significant future value as quantum hardware evolves, potentially providing computational advantages in more complex or high-dimensional tasks.

Future research efforts should focus on several key areas. Optimizing KAN-based architectures and exploring their scalability to larger and more complex datasets can further validate their practicality. At the same time, improving the robustness of QCNNs to noise and hardware imperfections is essential, possibly through the development of novel quantum error mitigation strategies. In

addition, broader comparative analyses of different data sets, such as medical or multispectral imaging, could further clarify the application domain of each technology. Investigation of hybrid architectures that strategically combine CNN, KAN, and quantum elements could provide synergistic benefits by leveraging their individual strengths.

Acknowledgments. This work was funded by the European Union - NextGenerationEU under the project NRRP (i)"National Centre for HPC, Big Data and Quantum Computing (HPC)" CN00000013 (CUP D43C22001240001) [MUR Decree n. 1031-17/06/2022] - Cascade Call launched by SPOKE 10 POLIMI: "QML-NTED" project, (ii) "National Quantum Science & Technology Institute (NQSTI)" PE00000023 (CUP B53C22004180005) [MUR Decree n. 341 15/03/2022] – Cascade Call launched by SPOKE 8 CNR: "QUANTIC" project. EU-FESR, PON Ricerca e Innovazione 2014–2020-DM 1062/2021.

The experiments have been performed by using the computing resources operated by the Department of Mathematics and Physics of the University of Campania "Luigi Vanvitelli", Caserta, Italy, within the VALERE Program.

References

1. Cheon, M.: Kolmogorov-arnold network for satellite image classification in remote sensing. arXiv preprint arXiv:2406.00600 (2024)
2. Cong, I., Choi, S., Lukin, M.D.: Quantum convolutional neural networks. Nat. Phys. **15**(12), 1273–1278 (2019)
3. Goodfellow, I.: Deep learning (2016)
4. Goodfellow, I.J., Shlens, J., Szegedy, C.: Explaining and harnessing adversarial examples. arXiv preprint arXiv:1412.6572 (2014)
5. Gupta, P., Saxena, N., Sharma, M., Tripathi, J.: Deep neural network for human face recognition. Int. J. Eng. Manuf. (IJEM) **8**(1), 63–71 (2018)
6. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
7. Huang, G.B., Mattar, M., Berg, T., Learned-Miller, E.: Labeled faces in the wild: a database for studying face recognition in unconstrained environments. In: Workshop on Faces in'Real-Life'Images: Detection, Alignment, and Recognition (2008)
8. Huang, S.Y., An, W.J., Zhang, D.S., Zhou, N.R.: Image classification and adversarial robustness analysis based on hybrid quantum–classical convolutional neural network. Optics Commun. **533** (2023). https://doi.org/10.1016/j.optcom.2023.129287
9. Hur, T., Kim, L., Park, D.K.: Quantum convolutional neural network for classical data classification. Quantum Mach. Intell. **4**(1), 3 (2022)
10. Kingma, D.P.: Adam: a method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
11. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems, vol. 25 (2012)
12. LeCun, Y., Bengio, Y., et al.: Convolutional networks for images, speech, and time series. The Handbook of Brain Theory and Neural Networks **3361**(10), 1995 (1995)

13. Liu, Z., et al.: KAN: Kolmogorov-Arnold Networks. arXiv preprint arXiv:2404.19756 (2024)
14. Rawat, W., Wang, Z.: Deep convolutional neural networks for image classification: a comprehensive review. Neural Comput. **29**(9), 2352–2449 (2017)
15. Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Learning representations by back-propagating errors. Nature **323**(6088), 533–536 (1986)
16. Schuld, M., Petruccione, F.: Machine Learning With Quantum Computers, vol. 676. Springer (2021). https://doi.org/10.1007/978-3-030-83098-4
17. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
18. Smaldone, A.M., Kyro, G.W., Batista, V.S.: Quantum convolutional neural networks for multi-channel supervised learning. Quantum Mach. Intell. **5**(2) (2023). https://doi.org/10.1007/s42484-023-00130-3