

A Quantum 3D Convolutional Neural Network with Application in Video Classification

Konstandinos Blekos and Dimitrios Kosmopoulos

University of Patras, Rio GR 26500, Greece
eelvex@gmail.com, dkosmo@upatras.gr

Abstract. Quantum computing seeks to exploit the properties of quantum mechanics to perform computations at a fraction of the cost compared to the classical computing methods. Recently, quantum methods for machine learning have attracted the interest of researchers. Those methods aim to exploit, in the context of machine learning, the potential benefits that the quantum computers should be able to offer in the near future. A particularly interesting area of research in this direction, investigates the union of quantum machine learning models with Convolutional Neural Networks. In this paper we develop a quantum counterpart of a 3D Convolutional Neural Network for video classification, dubbed Q3D-CNN. This is the first approach for quantum video classification we are aware of.

Our model is based on previously proposed quantum machine learning models, where manipulation of the input data is performed in such a way that a fully quantum-mechanical neural network layer can be realized and used to form a Quantum Convolutional Neural Network. We augment this approach by introducing quantum-friendly operations during data-loading and appropriately manipulating the quantum network. We demonstrate the applicability of the proposed Q3D-CNN in video classification using videos from a publicly available dataset. We successfully classify the test dataset using two and three classes using the quantum network and its classical counterpart.

1 Introduction

Quantum computing (QC) is using the as-yet-untapped quantum mechanical properties of nature, such as superposition and entanglement, to provide a new toolbox for computational problems. Such a toolbox is expected to provide great theoretical and technological advancements in the near future. A large number of algorithms have been proposed lately and are being investigated, while researchers are still trying to understand the advantages that a quantum computer has to offer.

Image processing algorithms are an important subset of algorithms where quantum computing is developing rapidly. Recently, in the context of image processing and computer vision research in general, there have been many interesting proposals for quantum processing techniques which employ fully-quantum,

quantum-classical hybrids or quantum-inspired models [8, 19, 12, 32, 10, 11]. On the other hand, to our best knowledge, the closely related subfield of video processing has not employed QC yet. During the last decade the deep learning methods have revolutionized the field of video classification using CNNs (see, e.g., [14], [30], [26], [6]) or RNNs (e.g., [31], [29], [27]). The research question we tackle in this paper is how to exploit the benefits of QC using a CNN architecture for video classification. This is a very challenging task but there are already many directions that are being investigated [9, 1, 3, 22, 21, 11, 16, 20, 5, 13]. A particularly interesting line of investigation is to directly translate neural network layers to quantum devices exploiting the better-understood methods of the classical ML algorithms [23, 4, 15, 2, 28, 24].

To the best of our knowledge, there has been no work for video processing using quantum machine learning techniques. Here we propose an experimental setup and we investigate the quantum video processing prospects. Thus we test the applicability of a quantum 3D-CNN (dubbed Q3D-CNN) for video classification.

Our main contributions are:

- a quantum machine learning procedure for video classification;
- a comparison of the proposed Q3D-CNN to the classical 3D-CNN with the same structure, and
- an investigation of the scaling properties of the quantum algorithm, thus highlighting the differences in efficiency to known classical algorithms

2 Related Work

Many models have been proposed lately for the extension or enhancement of Neural Networks using quantum computing techniques [13]. Few of the proposed models for quantum neural networks aim to function in a way similar to convolutional neural networks so that they can be useful in computer vision. A non-exhaustive list of these relevant quantum neural network architectures include Quantvolutional Neural Networks [11], Quantum Convolutional Neural Networks [7], Quantum M-P Neural Networks [34], Quantum Competitive Neural Network [33] and more. Variational Circuits represent a very important class of hybrid quantum-classical algorithms [5] that also implements neural networks. A Variational Neural Network (eg [25, 17]) is a parametrized quantum circuit, with the parameters been fed to a classical machine learning algorithm.

In [7] Cong et.al proposed a new quantum circuit model dubbed “Quantum Convolutional Neural Network” that could be used in signal processing. Their proposed quantum circuit model shares aspects with a classical CNN but can not reproduce in general the operations of a classical CNN and, therefore, can not be used to “translate” abstract classical CNN architectures to quantum.

At the same time, a more general approach was proposed by Kerenidis et.al [15]. Using the observation that an inner product estimation could replace the matrix convolution process of a CNN, they proposed a “hardware agnostic” method to construct a quantum CNN layer that can be used to build quantum

counterparts of any classical CNN architecture. This model is presented in detail in the next section. Our Q3D-CNN builds on top of this model so that it can be used for video classification.

3 Methodology

The main goal of this work is to build a quantum 3D-CNN, in such a way that (a) it can successfully discriminate different classes of video input data and (b) it can do so in a more efficient way than the equivalent classical CNN. To this end we first employ quantum-efficient replacements for the classical components of the 3D-CNN as proposed in [15]. Then, we add a quantum process with no efficient classical analog that further boosts the efficiency of the QCNN.

3.1 Quantum background

The whole point of “translating” a classical process to quantum is to try and take advantage of efficiency boosts that are offered by some quantum processes. Before describing these boosts and the way we use them, we will first briefly describe what it means to have a quantum version of a classical algorithm.

In a quantum computing process instead of using bits of 0 and 1 to store information we use quantum states of two levels (a *qubit*). These quantum states are 2D vectors of complex parameters and measure 1. A series of n quantum bits (often referred to as a *quantum register*) form a 2^n -dimensional vector of complex parameters and measure 1. The *computational basis*, then, is the one-hot orthonormal basis: $|0\rangle = (1, 0, \dots, 0)^T$, $|1\rangle = (0, 1, \dots, 0)^T$, $|2\rangle = \dots$.

The quantum states are manipulated by use of quantum gates which act as the quantum analog of the classical logical gates (AND, OR, NOT, etc); they are represented by complex unitary matrices. By acting the quantum gates on the quantum states, we can rebuild the classical logical circuits. The key differences of the quantum-vs-classical can be derived from this vector-matrices representation and can be summarized as follows:

- We can form and then exploit interference patterns when combining complex vectors in order to get to a result more efficiently than what is classically possible. This inference-pattern-exploitation is what is sometimes referred to as “taking advantage of quantum parallelism”.
- Applying unitary matrices on complex vectors is a linear and reversible process, therefore quantum computing is linear and reversible in nature; this forced linearity is a serious obstacle in implementing the many nonlinear processes of neural networks.
- A direct consequence of the previous point is that a quantum register (a series of quantum bits) can not be copied. This is a point that should be emphasized: there can be no physical way of copying a qubit. The exact complex parameters of a qubit are *unknowable*. Therefore, to extract the information from a quantum register one has to repeat the quantum algorithm while sampling the output.

When simulating a quantum computer, the output quantum state vector is obviously known. However, since the actual state is unknowable, when using a real quantum computer we can only approximate the output quantum state vector to an arbitrary degree by repeating the quantum process many times and keeping track of the distribution of the outcomes. This process is called *tomography* and has to be taken into consideration by inserting “noise” parameters into the simulation [15].

Another important issue is the conversion between classical and quantum data. Since the algorithms that we are concerned with deal with both classical and quantum procedures, a way to translate between classical and quantum data is needed. For example, the input image is initially stored in a classical computer but needs to be converted to a quantum state so that can be manipulated by the quantum CPU. A usual practice, and the one that we use in this paper, is to encode a classical vector by mapping its elements to the corresponding amplitudes of the basis vectors. So, if $v = (v_0, v_1, \dots, v_n)$:

$$|v\rangle = \frac{1}{\|v\|} \sum v_i |i\rangle$$

Using this encoding, one can devise classical data structures (a “quantum RAM” (QRAM)) that provide efficient implementations for the crucial quantum state storing and retrieving procedures.

3.2 Quantum Convolutional Neural Network

We now turn to the implementation of the Q3D-CNN. The crucial parts of a classical CNN are the input matrices, the kernels, the convolution between them and the nonlinear activation function. Ideally, we would have an at-least-as-efficient quantum counterpart for each of these parts. We construct a quantum CNN based on the procedure in [15]. Our key differences are a) we are applying the algorithm to 3D volumes of videos instead of greyscale images and, most importantly, b) we introduce a quantum-efficient preprocessing step based on the observation that video classification heavily depends on differences between successive frames. This is a crucial step that significantly boosts the efficiency of the network as we will show.

The key observation in [15] is that the matrix convolution can be replaced by the quantum-efficient *inner product estimation algorithm*. The inner product estimation algorithm estimates inner products between two quantum vectors with high probability and high efficiency. To use the algorithm, the image and kernel matrices have first to be unraveled so as regions where matrix multiplication was to be applied, are now rows and columns where inner product operation will be applied. More specifically, a region $w \times h$ of the input image is converted to a quantum vector $|A\rangle$ and the corresponding $w \times h$ kernel is converted to a quantum vector $|F\rangle$. The inner product $\langle A|F\rangle$ is then estimated using the efficient quantum routine (Figure 1). If the matrices are stored in a QRAM, the rows and columns can be directly efficiently extracted.

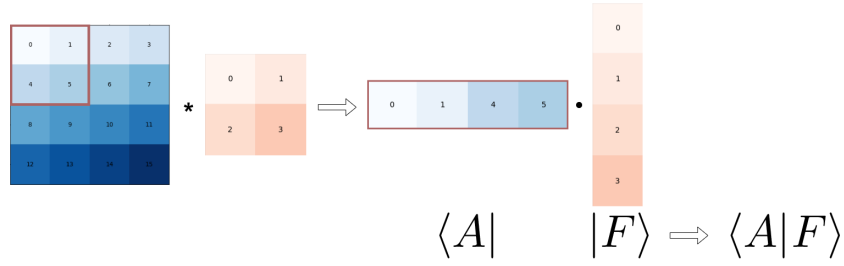


Fig. 1. Converting matrix convolution to inner product operation.

The difference between successive frames can be calculated quantum-efficiently for vectors that are stored in a QRAM at the expense of using one more qubit (an *ancilla* qubit). For two successive frames f_0, f_1 we can form the following quantum stating using only two QRAM queries

$$|f_0\rangle |0\rangle_a + |f_1\rangle |1\rangle_a$$

Then, by applying only one quantum operation on the ancilla qubit and measuring until we find the ancilla qubit in state $|1\rangle_a$ we are left with the state $|f_0\rangle - |f_1\rangle$ which represents the difference between two successive frames.

To form all difference frames, we can either store the new state as the input frame and repeat the process for the rest of the frames, or, if the number of qubits is a cheaper resource, we can form all difference-frames with just one pass by using one ancilla qubit for each new frame (Figure 2).

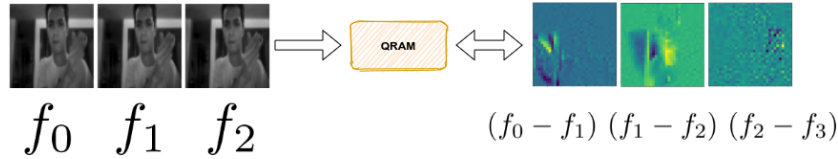


Fig. 2. Process of calculating and storing the difference frames

The full algorithm, for a single layer, works as follows (see Algorithm 1; step numbers referred in Figure 3):

- 1 Store into QRAM: the classical matrices that represent the input image layers and the kernels are stored in an efficient QRAM structure.
- 1A “3D” preprocess: Perform the difference operation and update the QRAM.
- 2 Unravel: from the QRAM we can efficiently extract quantum states that represent regions of the input image layers as rows and the kernels as columns.
- 3 Inner Product Estimation: Using the inner product estimation algorithm we construct a state that is proportional to the inner product of the rows and

columns of the previous step. This represents the convolution of the initial input images and kernels.

- 4 Activation function and QRAM update: A final step applies a classical non-linear activation function to the inner product while updating the QRAM by sampling the output of the previous step.

The next layer of a QCNN loops back from step 2 as many times as there are layers to the CNN.

Algorithm 1 Q3D-CNN layers

```

QRAM  $\leftarrow$  (video, kernels)
Perform difference operation  $\rightarrow$  QRAM
repeat
  Unravel and perform Inner Product Estimation
  Perform the classical operations while updating the QRAM
until no more layers
  
```

4 Experimental evaluation

We now describe an implementation of the Q3D-CNN that showcases the network performance and the accuracy advantage that the difference-operation provides. In the following we describe the dataset, the network architectures and report the respective classification results.

Dataset and pre-processing We evaluate our Q3D-CNN using a small subset of the publicly available *20BN-jester Dataset V1*, containing labeled video clips showing humans performing predefined hand gestures [18]. The full dataset contains about 150000 video samples split in 27 classes. Each video has a height of 100 pixels and variable width of the same order. The average duration of each video sample is around 36 frames at 12 fps. To be able to perform the training simulations at a reasonable time we used only a small part of the dataset and we reduced those videos both in dimensions and duration. We cropped and downscaled each image keeping only the Red channel (see Table 1).

We perform experiments with two and three of the available classes for training. Preliminary runs have shown that the method scales well for more classes in terms of accuracy but greatly increases the simulation times so we opted for the smaller classes as proof-of-concept work.

Network Architecture We trained four different networks for two versions of the dataset as shown in Table 2. For each dataset, comprised by two or three classes, we run the classical and quantum CNN with or without the 3D preprocessing step. Different input sizes were tried with similar results. We only present here the smaller (32×32)-sized datasets, as these better highlight the accuracy advantage given by the 3D step.

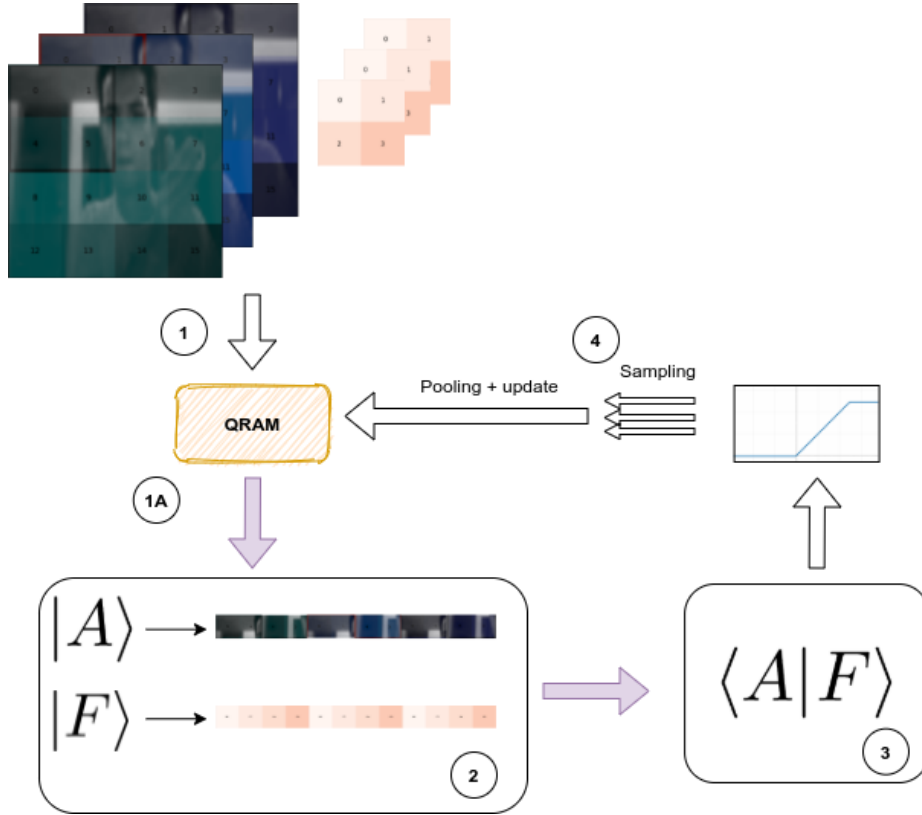


Fig. 3. Graphical summary of a quantum convolutional layer [15]

The network architecture, common among all instances, was kept as simple as possible so as the results can be as comparable as possible though, as already noted, a truly direct comparison cannot be made. We used two convolutional layers with 8×8 kernels followed by two fully connected layers and one output layer. The activation function is a capped ReLu [15] with a cap at 10.

Training. Both the classical and quantum networks were trained for 200 epochs using a simple backpropagation algorithm with learning rate of 0.01. We use the quantum version of the backpropagation algorithm as described by [15].

Simulation and simulation parameters. Since there are as yet no real quantum computer implementations, the quantum processes have to be simulated on a classical computer. The simulation requires sampling of the output of the quantum processes as described in Section 3.1. This greatly affects the accuracy of the network and execution speed. It is obvious that until there is a real quantum computing device to run the quantum algorithms on, the evaluation of the algorithms is incomplete at best. For a more complete discussion of the simulation parameters we refer the reader to [15].

Label	Videos	Image size	Frames
Swipe Left	250	32×32	16
Swipe Right	250	32×32	16
Turning Hand Clockwise	250	32×32	16

Table 1. The reduced 20BN-Jester dataset.

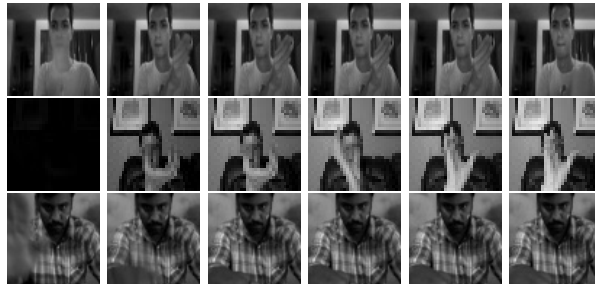


Fig. 4. Sample frames for the classes of “Swipe Left” (top row), “Turning Hand Clockwise” (middle row) and “Swipe Down” (bottom row).

Results. We evaluate the performance of the Q3D-CNN in terms of prediction accuracy and run-time efficiency. Since we can not directly compare training efficient and run-times—as quantum devices for tasks like this don’t yet exist—we report the calculated complexity of the networks. We compare each of the four networks’ performance (CNN, 3D-CNN, QCNN and Q3D-CNN) for each dataset (2 & 3 classes) (Table 2). The corresponding confusion matrices are shown in Table 3.

We see that the best accuracy was achieved by the 3D-CNN at 87% for 2 classes and 72% for three classes. The accuracies for the Q3D-CNN were very similar to the classical with 83% and 67% respectively. The non-3D versions of both networks achieved, as was expected, significantly lower accuracy values, validating the use of the “3D” post-processing for improving the network’s performance.

In terms of network complexity, we highlight the difference in efficiency between the classical and quantum “3D” post-processing operation. We notate \tilde{O}_C and \tilde{O}_Q the base classical and quantum network complexities respectively. In general, \tilde{O}_C depends on the kernel and input image sizes quadratically, while \tilde{O}_Q depends on the kernel and input image sizes almost linearly. For a more detailed analysis of \tilde{O}_Q see [15]. The “3D” operation, on the other hand, has a complexity of $O(\text{size})$ for the classical case but a complexity of $O(1)$ for the quantum case, as the depth of the quantum circuit needed to perform the operation is constant.

Network	Classes	Accuracy	Complexity
CNN	2	0.69	\tilde{O}_C
	3	0.49	
QCNN	2	0.73	\tilde{O}_Q
	3	0.58	
3D-simulated CNN	2	0.87	$O(\text{size}) + \tilde{O}_C$
	3	0.72	
Q3D-CNN	2	0.83	$O(1) + \tilde{O}_Q$
	3	0.67	

Table 2. Comparison of accuracy and one-layer-complexity for the various classical and quantum networks.

CNN Predicted			CNN			Predicted		
A	C		A	B	C	A	B	C
True A	0.30	0.16	True A	0.10	0.06	0.15		
			True B	0.00	0.17	0.15		
True C	0.15	0.39	True C	0.04	0.11	0.22		
QCNN Predicted			QCNN			Predicted		
A	C		A	B	C	A	B	C
True A	0.25	0.23	True A	0.17	0.08	0.07		
			True B	0.02	0.22	0.11		
True C	0.04	0.48	True C	0.07	0.08	0.19		
3D-CNN Predicted			3D-CNN			Predicted		
A	C		A	B	C	A	B	C
True A	0.43	0.06	True A	0.23	0.03	0.03		
			True B	0.02	0.22	0.09		
True C	0.06	0.44	True C	0.03	0.08	0.27		
Q3D-CNN Predicted			Q3D-CNN			Predicted		
A	C		A	B	C	A	B	C
True A	0.38	0.11	True A	0.20	0.05	0.04		
			True B	0.03	0.21	0.10		
True C	0.06	0.45	True C	0.03	0.08	0.26		

Table 3. Confusion matrices for 2- and 3-class datasets and all networks. A = “Swiping Left”, B = “Swiping Down”, C = “Turning Hand Clockwise”

5 Discussion and Conclusion

We presented a convolutional neural network architecture built using quantum processes that is able to discriminate between different classes of videos with comparable accuracy to the classical counterpart but with higher efficiency. This is the first approach for quantum video classification we are aware of. It is based on quantum-efficiently calculating the difference between successive video frames

and then training a quantum convolutional neural network by replacing the convolution operation with a quantum inner product estimation [15].

The classification performance of the Q3D-CNN appears to be a bit lower compared to the classical one. However, the trade-off between accuracy and efficiency that our Q3D-CNN offers, appears to be an option worth considering, especially in the context of time-critical applications. We should note here that the Q3D-CNN becomes more efficient, compared to the 3D-CNN, as the kernel and layer size increase, meaning that it could provide an attractive starting point for quantum video processing.

Furthermore, the process of calculating the difference between the successive video frames provides two advantages. Firstly, it significantly boosts the prediction accuracy in both networks (59% mean to 80% for the classical networks and 66% to 79% for the quantum versions). Secondly it is a very quantum-efficient operation requiring only a constant number of quantum operations to perform, it is independent of the input size, at the expense of a few more qubits.

ACKNOWLEDGMENT This research has been co-financed by the European Regional Development Fund of the European Union and Greek national funds through the Operational Program "Competitiveness, Entrepreneurship and Innovation", under the call "RESEARCH – CREATE – INNOVATE" (project code:T1EDK-0982)

References

1. Adcock, J., Allen, E., Day, M., Frick, S., Hinchliff, J., Johnson, M., Morley-Short, S., Pallister, S., Price, A., Stanic, S.: Advances in quantum machine learning. arXiv:1512.02900 [quant-ph] (Dec 2015)
2. Allcock, J., Hsieh, C.Y., Kerenidis, I., Zhang, S.: Quantum Algorithms for Feedforward Neural Networks. *ACM Transactions on Quantum Computing* **1**(1), 6:1–6:24 (Oct 2020). <https://doi.org/10.1145/3411466>
3. Allcock, J., Zhang, S.: Quantum machine learning. *National Science Review* **6**(1), 26–28 (Jan 2019). <https://doi.org/10.1093/nsr/nwy149>
4. Behrman, E.C., Nash, L.R., Steck, J.E., Chandrashekar, V.G., Skinner, S.R.: Simulations of quantum neural networks. *Information Sciences* **128**(3), 257–269 (Oct 2000). [https://doi.org/10.1016/S0020-0255\(00\)00056-6](https://doi.org/10.1016/S0020-0255(00)00056-6)
5. Cerezo, M., Arrasmith, A., Babbush, R., Benjamin, S., Endo, S., Fujii, K., McClean, J.R., Mitarai, K., Yuan, X., Cincio, L., Coles, P.: Variational Quantum Algorithms. arXiv:2012.09265 (2020)
6. Chatzis, S.P., Kosmopoulos, D.: A nonparametric bayesian approach toward stacked convolutional independent component analysis. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (Dec 2015)
7. Cong, I., Choi, S., Lukin, M.D.: Quantum Convolutional Neural Networks. *Nature Physics* **15**(12), 1273–1278 (Dec 2019). <https://doi.org/10.1038/s41567-019-0648-8>
8. Dang, Y., Jiang, N., Hu, H., Ji, Z., Zhang, W.: Image classification based on quantum K-Nearest-Neighbor algorithm. *Quantum Information Processing* **17**(9), 239 (Aug 2018). <https://doi.org/10.1007/s11128-018-2004-9>
9. Garg, S., Ramakrishnan, G.: *Advances in Quantum Deep Learning: An Overview*. arXiv:2005.04316 [quant-ph] (May 2020)

10. Gawron, P., Lewiński, S.: Multi-Spectral Image Classification with Quantum Neural Network. In: IGARSS 2020 - 2020 IEEE International Geoscience and Remote Sensing Symposium. pp. 3513–3516 (Sep 2020). <https://doi.org/10.1109/IGARSS39084.2020.9323065>
11. Henderson, M., Shakya, S., Pradhan, S., Cook, T.: Quvolutional neural networks: Powering image recognition with quantum circuits. *Quantum Machine Intelligence* **2**(1), 1–9 (Jun 2020). <https://doi.org/10.1007/s42484-020-00012-y>
12. Hernández, H.I.G., Ruiz, R.T., Sun, G.H.: Image Classification via Quantum Machine Learning. arXiv:2011.02831 [quant-ph] (Dec 2020)
13. Jeswal, S.K., Chakraverty, S.: Recent Developments and Applications in Quantum Neural Network: A Review. *Archives of Computational Methods in Engineering* **26**(4), 793–807 (Sep 2019). <https://doi.org/10.1007/s11831-018-9269-0>
14. Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., Fei-Fei, L.: Large-scale video classification with convolutional neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (Jun 2014)
15. Kerenidis, I., Landman, J., Prakash, A.: Quantum Algorithms for Deep Convolutional Neural Networks. In: International Conference on Learning Representations (Sep 2019)
16. Kulkarni, V., Kulkarni, M., Pant, A.: Quantum Computing Methods for Supervised Learning. arXiv:2006.12025 [quant-ph] (Jun 2020)
17. Lockwood, O., Si, M.: Reinforcement Learning with Quantum Variational Circuits. arXiv:2008.07524 [quant-ph, stat] (Aug 2020)
18. Materzynska, J., Berger, G., Bax, I., Memisevic, R.: The jester dataset: A large-scale video dataset of human gestures. In: 2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW). pp. 2874–2882. IEEE Computer Society (2019)
19. Nguyen, N.T., Kenyon, G.T.: Image Classification Using Quantum Inference on the D-Wave 2X. In: 2018 IEEE International Conference on Rebooting Computing (ICRC). pp. 1–7 (Nov 2018). <https://doi.org/10.1109/ICRC.2018.8638596>
20. Niu, X.F., Ma, W.P.: A novel quantum neural network based on multi-level activation function. *Laser Physics Letters* **18**(2), 025201 (Jan 2021). <https://doi.org/10.1088/1612-202X/abd23c>
21. Oh, S., Choi, J., Kim, J.: A Tutorial on Quantum Convolutional Neural Networks (QCNN). arXiv:2009.09423 [quant-ph] (Sep 2020)
22. Perdomo-Ortiz, A., Benedetti, M., Realpe-Gómez, J., Biswas, R.: Opportunities and challenges for quantum-assisted machine learning in near-term quantum computers. *Quantum Science and Technology* **3**(3), 030502 (Jun 2018). <https://doi.org/10.1088/2058-9565/aab859>
23. Schuld, M., Sinayskiy, I., Petruccione, F.: Simulating a perceptron on a quantum computer. *Physics Letters A* **379**(7), 660–663 (Mar 2015). <https://doi.org/10.1016/j.physleta.2014.11.061>
24. Tacchino, F., Barkoutsos, P., Macchiavello, C., Tavernelli, I., Gerace, D., Bajoni, D.: Quantum implementation of an artificial feed-forward neural network. *Quantum Science and Technology* **5**(4), 044010 (Oct 2020). <https://doi.org/10.1088/2058-9565/abb8e4>
25. Tacchino, F., Barkoutsos, P.K., Macchiavello, C., Gerace, D., Tavernelli, I., Bajoni, D.: Variational learning for quantum artificial neural networks. In: 2020 IEEE International Conference on Quantum Computing and Engineering (QCE). pp. 130–136 (Oct 2020). <https://doi.org/10.1109/QCE49297.2020.00026>

26. Tran, D., Wang, H., Torresani, L., Feiszli, M.: Video classification with channel-separated convolutional networks. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) (Oct 2019)
27. Ullah, A., Ahmad, J., Muhammad, K., Sajjad, M., Baik, S.W.: Action recognition in video sequences using deep bi-directional LSTM with CNN features. *IEEE Access* **6**, 1155–1166 (2018). <https://doi.org/10.1109/ACCESS.2017.2778011>
28. Wan, K.H., Dahlsten, O., Kristjánsson, H., Gardner, R., Kim, M.S.: Quantum generalisation of feedforward neural networks. *npj Quantum Information* **3**(1), 1–8 (Sep 2017). <https://doi.org/10.1038/s41534-017-0032-4>
29. Wu, Z., Wang, X., Jiang, Y.G., Ye, H., Xue, X.: Modeling spatial-temporal clues in a hybrid deep learning framework for video classification. In: Proceedings of the 23rd ACM International Conference on Multimedia. pp. 461–470. MM '15, Association for Computing Machinery, New York, NY, USA (2015). <https://doi.org/10.1145/2733373.2806222>
30. Xie, S., Sun, C., Huang, J., Tu, Z., Murphy, K.: Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification. In: Proceedings of the European Conference on Computer Vision (ECCV) (Sep 2018)
31. Yue-Hei Ng, J., Hausknecht, M., Vijayanarasimhan, S., Vinyals, O., Monga, R., Toderici, G.: Beyond short snippets: Deep networks for video classification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (Jun 2015)
32. Zhou, N.R., Liu, X.X., Chen, Y.L., Du, N.S.: Quantum K-Nearest-Neighbor Image Classification Algorithm Based on K-L Transform. *International Journal of Theoretical Physics* **60**(3), 1209–1224 (Mar 2021). <https://doi.org/10.1007/s10773-021-04747-7>
33. Zhou, R.: Quantum Competitive Neural Network. *International Journal of Theoretical Physics* **49**(1), 110 (Nov 2009). <https://doi.org/10.1007/s10773-009-0183-y>
34. Zhou, R., Ding, Q.: Quantum M-P Neural Network. *International Journal of Theoretical Physics* **46**(12), 3209–3215 (Dec 2007). <https://doi.org/10.1007/s10773-007-9437-8>